Package ‘robCompositions’

September 20, 2021

Type Package
Title Compositional Data Analysis
Version 2.3.1
Date 2021-09-13
Depends R (>= 3.5.0), ggplot2, pls, data.table
LinkingTo Rcpp, RcppEigen
Imports car, cvTools, e1071, fda, rrcov, cluster, fpc, GGally,
    ggfortify, kernlab, MASS, mclust, tidyr, robustbase, robustHD,
    splines, VIM, zCompositions, reshape2, Rcpp
Suggests knitr, testthat
VignetteBuilder knitr
Maintainer Matthias Templ <matthias.templ@gmail.com>
Description Methods for analysis of compositional data including robust
    methods (doi:10.1007/978-3-319-96422-5), imputation of missing values (doi:10.1016/j.csda.2009.11.023),
    methods to deal with essential zeros (doi:10.1080/02664763.2016.1182135), (robust) outlier
detection for compositional data, (robust) principal component analysis for
    compositional data, (robust) factor analysis for compositional data, (robust) discriminant analysis for compositional data (Fisher rule), robust regression
    with compositional predictors, functional data analy-
    sis (doi:10.1016/j.csda.2015.07.007) and p-splines (doi:10.1016/j.csda.2015.07.007),
    contingency (doi:10.1080/03610926.2013.824980)
    and compositional ta-
    and (robust) Anderson-Darling normality tests for
    compositional data as well as popular log-ratio transformations (addLR, cenLR,
    isomLR, and their inverse transformations). In addition, visualisation and
diagnostic tools are implemented as well as high and low-level plot functions
for the ternary diagram.
License GPL (>= 2)
LazyLoad: yes
LazyData: true
Encoding: UTF-8
RoxygenNote: 7.1.1
NeedsCompilation: yes

Author: Matthias Templ [aut, cre] (<https://orcid.org/0000-0002-8638-5276>),
        Karel Hron [aut] (<https://orcid.org/0000-0002-1847-6598>),
        Peter Filzmoser [aut] (<https://orcid.org/0000-0002-8014-4682>),
        Kamila Facevicova [ctb],
        Petra Kynclova [ctb],
        Jan Walach [ctb],
        Veronika Pintar [ctb],
        Jiajia Chen [ctb],
        Dominika Miksova [ctb],
        Bernhard Meindl [ctb],
        Alessandra Menafoglio [ctb] (<https://orcid.org/0000-0003-0682-6412>),
        Alessia Di Blasi [ctb],
        Federico Pavone [ctb],
        Nikola Stefelova [ctb],
        Gianluca Zeni [ctb]

Repository: CRAN

Date/Publication: 2021-09-20 14:20:05 UTC

### R topics documented:

- robCompositions-package .............................................. 5
- addLR .......................................................... 7
- addLRinv ...................................................... 9
- aDist ......................................................... 10
- adjust .......................................................... 12
- adtest ....................................................... 13
- adtestWrapper .................................................. 14
- ageCatWorld ..................................................... 16
- alcohol ......................................................... 17
- alcoholreg ...................................................... 18
- arcticLake ...................................................... 18
- balances ....................................................... 19
- biomarker ....................................................... 21
- biplot.factanal .................................................. 23
- biplot.pcaCoDa .................................................. 24
- bootnComp ...................................................... 25
- cancer ......................................................... 26
- cancerMN ....................................................... 27
- ced ............................................................ 28
- cenLR .......................................................... 29
- cenLRinv ...................................................... 30
<table>
<thead>
<tr>
<th>R topics documented:</th>
</tr>
</thead>
<tbody>
<tr>
<td>chrorizonDL</td>
</tr>
<tr>
<td>clustCoDa</td>
</tr>
<tr>
<td>clustCoDa_qmode</td>
</tr>
<tr>
<td>coffee</td>
</tr>
<tr>
<td>compareMahal</td>
</tr>
<tr>
<td>compositionalSpline</td>
</tr>
<tr>
<td>constSum</td>
</tr>
<tr>
<td>coord</td>
</tr>
<tr>
<td>corCoDa</td>
</tr>
<tr>
<td>cubeCoord</td>
</tr>
<tr>
<td>daCoDa</td>
</tr>
<tr>
<td>daFisher</td>
</tr>
<tr>
<td>economy</td>
</tr>
<tr>
<td>educFM</td>
</tr>
<tr>
<td>efsa</td>
</tr>
<tr>
<td>election</td>
</tr>
<tr>
<td>electionATbp</td>
</tr>
<tr>
<td>employment</td>
</tr>
<tr>
<td>employment2</td>
</tr>
<tr>
<td>employment_df</td>
</tr>
<tr>
<td>expenditures</td>
</tr>
<tr>
<td>expendituresEU</td>
</tr>
<tr>
<td>fcenLR</td>
</tr>
<tr>
<td>fcenLRinv</td>
</tr>
<tr>
<td>fcenLRp</td>
</tr>
<tr>
<td>fcenLRu</td>
</tr>
<tr>
<td>foodbalance</td>
</tr>
<tr>
<td>GDPsatis</td>
</tr>
<tr>
<td>gemas</td>
</tr>
<tr>
<td>gjovik</td>
</tr>
<tr>
<td>gm</td>
</tr>
<tr>
<td>gmean_sum</td>
</tr>
<tr>
<td>goexp</td>
</tr>
<tr>
<td>haplogroups</td>
</tr>
<tr>
<td>honey</td>
</tr>
<tr>
<td>ilr.2x2</td>
</tr>
<tr>
<td>impAll</td>
</tr>
<tr>
<td>impCoda</td>
</tr>
<tr>
<td>impKNNa</td>
</tr>
<tr>
<td>impRZalr</td>
</tr>
<tr>
<td>impRZilr</td>
</tr>
<tr>
<td>imputeBDLs</td>
</tr>
<tr>
<td>imputeUDLs</td>
</tr>
<tr>
<td>ind2x2</td>
</tr>
<tr>
<td>indTab</td>
</tr>
<tr>
<td>instw</td>
</tr>
<tr>
<td>int2x2</td>
</tr>
<tr>
<td>intArray</td>
</tr>
</tbody>
</table>
### R topics documented:

- `intTab` .................................................................................. 96
- `is.equivalent` ........................................................................ 97
- `isic32` .................................................................................. 98
- `laborForce` ........................................................................... 98
- `lifeExpGdp` .......................................................................... 99
- `lmCoDaX` .............................................................................. 100
- `machineOperators` ................................................................. 101
- `manu_abs` ............................................................................. 102
- `mcad` ................................................................................... 103
- `missPatterns` ......................................................................... 104
- `mortality` ............................................................................... 105
- `mortality_tab` ...................................................................... 106
- `nutrients` .............................................................................. 107
- `nutrients_branded` ............................................................... 109
- `orthbasis` ............................................................................. 110
- `outCoDa` ............................................................................... 111
- `payments` ............................................................................. 112
- `pcaCoDa` .............................................................................. 114
- `perturbation` ......................................................................... 116
- `pfa` ..................................................................................... 117
- `phd` ..................................................................................... 119
- `phd_totals` .......................................................................... 120
- `pivotCoord` .......................................................................... 121
- `plot.imp` ............................................................................. 123
- `plot.pcaCoDa` .................................................................... 125
- `plot.smoothSpl` .................................................................. 126
- `precipitation` ....................................................................... 127
- `print.imp` ........................................................................... 128
- `production` ........................................................................... 129
- `pTab` .................................................................................. 130
- `rcodes` ................................................................................. 131
- `rdcm` .................................................................................. 131
- `rSDev` .................................................................................. 132
- `rSDev.test` .......................................................................... 133
- `saffron` ............................................................................... 134
- `SDev` ................................................................................... 136
- `skyeLavas` ........................................................................... 137
- `smoothSplines` .................................................................... 138
- `smoothSplinesVal` ............................................................... 140
- `socExp` ................................................................................. 141
- `stats` ................................................................................... 142
- `summary.imp` ..................................................................... 144
- `tabCoord` ............................................................................. 145
- `teachingStuff` ...................................................................... 148
- `ternaryDiag` ....................................................................... 149
- `ternaryDiagAbline` ............................................................. 151
- `ternaryDiagEllipse` ............................................................ 152
- `ternaryDiagPoints` .............................................................. 153
Robust Estimation for Compositional Data.

Description

The package contains methods for imputation of compositional data including robust methods, (robust) outlier detection for compositional data, (robust) principal component analysis for compositional data, (robust) factor analysis for compositional data, (robust) discriminant analysis (Fisher rule) and (robust) Anderson-Darling normality tests for compositional data as well as popular log-ratio transformations (alr, clr, ilr, and their inverse transformations).

Details

Package: robCompositions
Type: Package
Version: 1.3.3
Date: 2009-11-28
License: GPL 2
LazyLoad: yes

Author(s)

Matthias Templ, Peter Filzmoser, Karel Hron,
Maintainer: Matthias Templ <templ@tuwien.ac.at>

References


Examples

```r
## k nearest neighbor imputation
data(expenditures)
expenditures[1,3]
expenditures[1,3] <- NA
impKNNa(expenditures)$xImp[1,3]

## iterative model based imputation
data(expenditures)
x <- expenditures
x[1,3]
x[1,3] <- NA
xi <- impCoda(x)$xImp
xi[1,3]
s1 <- sum(x[1,-3])
impS <- sum(xi[1,-3])
xi[,3] * s1/impS

xi <- impKNNa(expenditures)
xi
summary(xi)
## Not run: plot(xi, which=1)
plot(xi, which=2)
plot(xi, which=3)

## pca
data(expenditures)
p1 <- pcaCoDa(expenditures)
p1
plot(p1)

## outlier detection
data(expenditures)
oD <- outCoDa(expenditures)
oD
plot(oD)

## transformations
data(arcticLake)
```
x <- arcticLake
x.alr <- addLR(x, 2)
y <- addLRinv(x.alr)
addLRinv(addLR(x, 3))
data(expenditures)
x <- expenditures
y <- addLRinv(addLR(x, 5))
head(x)
head(y)
addLRinv(x.alr, ivar=2, useClassInfo=FALSE)

data(expenditures)
eclr <- cenLR(expenditures)
ineclr <- cenLRinv(eclr)
head(expenditures)
head(eineclr)
head(cenLRinv(eclr$x.clr))

require(MASS)
Sigma <- matrix(c(5.05,4.95,4.95,5.05), ncol=2, byrow=TRUE)
z <- pivotCoordInv(mvrnorm(100, mu=c(0,2), Sigma=Sigma))

---

**addLR**

**Additive logratio coordinates**

**Description**

The additive logratio coordinates map D-part compositional data from the simplex into a (D-1)-dimensional real space.

**Usage**

```
addLR(x, ivar = ncol(x), base = exp(1))
```

**Arguments**

- **x**: D-part compositional data
- **ivar**: Rationing part
- **base**: a positive or complex number: the base with respect to which logarithms are computed. Defaults to exp(1).

**Details**

The compositional parts are divided by the rationing part before the logarithm is taken.
Value

A list of class “alr” which includes the following content:

- `x.alr`: the resulting coordinates
- `varx`: the rationing variable
- `ivar`: the index of the rationing variable, indicating the column number of the rationing variable in the data matrix `x`
- `cnames`: the column names of `x`

The additional information such as `cnames` or `ivar` is useful when an inverse mapping is applied on the ‘same’ data set.

Author(s)

Matthias Templ

References


See Also

`addLRinv`, `pivotCoord`

Examples

data(arcticLake)
x <- arcticLake 
x.alr <- addLR(x, 2)
y <- addLRinv(x.alr)
## This exactly fulfills:
addLRinv(addLR(x, 3))

data(expenditures)
x <- expenditures 
y <- addLRinv(addLR(x, 5))
head(x)
head(y)
## --> absolute values are preserved as well.

## preserve only the ratios:
addLRinv(x.alr, ivar=2, useClassInfo=FALSE)
**addLRinv**

*Inverse additive logratio mapping*

### Description

Inverse additive logratio mapping, often called additive logistic transformation.

### Usage

```r
addLRinv(x, cnames = NULL, ivar = NULL, useClassInfo = TRUE)
```

### Arguments

- **x**: data set, object of class “alr”, “matrix” or “data.frame”
- **cnames**: column names. If the object is of class “alr” the column names are chosen from therein.
- **ivar**: index of the rationing part. If the object is of class “alr” the column names are chosen from therein. If not and ivar is not provided by the user, it is assumed that the rationing part was the last column of the data in the simplex.
- **useClassInfo**: if FALSE, the class information of object x is not used.

### Details

The function allows also to preserve absolute values when class info is provided. Otherwise only the relative information is preserved.

### Value

the resulting compositional data matrix

### Author(s)

Matthias Templ

### References


### See Also

`pivotCoordInv, cenLRinv, cenLR, addLR`
Examples

```r
data(arcticLake)
x <- arcticLake
x.alr <- addLR(x, 2)
y <- addLRinv(x.alr)
## This exactly fulfills:
addLRinv(addLR(x, 3))
data(expenditures)
x <- expenditures
y <- addLRinv(addLR(x, 5, 2))
head(x)
head(y)
## --> absolute values are preserved as well.

## preserve only the ratios:
addLRinv(x.alr, ivar=2, useClassInfo=FALSE)
```

---

**aDist**

_Aitchison distance_

**Description**

Computes the Aitchison distance between two observations, between two data sets or within observations of one data set.

**Usage**

```r
aDist(x, y = NULL)
iprod(x, y)
```

**Arguments**

- `x`  
a vector, matrix or data.frame
- `y`  
a vector, matrix or data.frame with equal dimension as `x` or `NULL`.

**Details**

This distance measure accounts for the relative scale property of compositional data. It measures the distance between two compositions if `x` and `y` are vectors. It evaluates the sum of the distances between `x` and `y` for each row of `x` and `y` if `x` and `y` are matrices or data frames. It computes a n times n distance matrix (with n the number of observations/compositions) if only `x` is provided.

The underlying code is partly written in C and allows a fast computation also for large data sets whenever `y` is supplied.
Value

The Aitchison distance between two compositions or between two data sets, or a distance matrix in case codey is not supplied.

Author(s)

Matthias Templ, Bernhard Meindl

References


See Also

pivotCoord

Examples

data(expenditures)
x <- xOrig <- expenditures
## Aitchison distance between two 2 observations:
aDist(x[1, ], x[2, ])
## Aitchison distance of x:
aDist(x)
## Example of distances between matrices:
## set some missing values:
## impute the missing values:
xImp <- impCoda(x, method="ltsReg")$xImp
## calculate the relative Aitchison distance between xOrig and xImp:
aDist(xOrig, xImp)

data("expenditures")
aDist(expenditures)
x <- expenditures[, 1]
y <- expenditures[, 2]
aDist(x, y)
aDist(expenditures, expenditures)
**adjust**

**Adjusting for original scale**

**Description**

Results from the model based iterative methods provides the results in another scale (but the ratios are still the same). This function rescale the output to the original scale.

**Usage**

`adjust(x)`

**Arguments**

- `x` object from class ‘imp’

**Details**

It is self-explaining if you try the examples.

**Value**

The object of class ‘imp’ but with the adjusted imputed data.

**Author(s)**

Matthias Templ

**References**


**See Also**

- `impCoda`

**Examples**

```r
data(expenditures)
x <- expenditures
xi <- impCoda(x)
x
xi$xImp
adjust(xi)$xImp
```
Description

This function provides three kinds of Anderson-Darling Normality Tests (Anderson and Darling, 1952).

Usage

adtest(x, R = 1000, locscatt = "standard")

Arguments

x         either a numeric vector, or a data.frame, or a matrix
R         Number of Monte Carlo simulations to obtain p-values
locscatt  standard for classical estimates of mean and (co)variance. robust for robust estimates using ‘covMcd()’ from package robustbase

Details

Three version of the test are implemented (univariate, angle and radius test) and it depends on the data which test is chosen.

If the data is univariate the univariate Anderson-Darling test for normality is applied.
If the data is bivariate the angle Anderson-Darling test for normality is performed out.
If the data is multivariate the radius Anderson-Darling test for normality is used.
If ‘locscatt’ is equal to “robust” then within the procedure, robust estimates of mean and covariance are provided using ‘covMcd()’ from package robustbase.

To provide estimates for the corresponding p-values, i.e. to compute the probability of obtaining a result at least as extreme as the one that was actually observed under the null hypothesis, we use Monte Carlo techniques where we check how often the statistic of the underlying data is more extreme than statistics obtained from simulated normal distributed data with the same (column-wise-) mean(s) and (co)variance.

Value

statistic     The result of the corresponding test statistic
method        The chosen method (univariate, angle or radius)
p.value       p-value

Note

These functions are use by adtestWrapper.
Author(s)

Karel Hron, Matthias Templ

References


See Also

adtestWrapper

Examples

```r
adtest(rnorm(100))
data(machineOperators)
x <- machineOperators
adtest(pivotCoord(x[,1:2]))
adtest(pivotCoord(x[,1:3]))
adtest(pivotCoord(x))
adtest(pivotCoord(x[,1:2]), locscatt="robust")
```

Description

A set of Anderson-Darling tests (Anderson and Darling, 1952) are applied as proposed by Aitchison (Aitchison, 1986).

Usage

```r
adtestWrapper(x, alpha = 0.05, R = 1000, robustEst = FALSE)
```

Arguments

- `x`: compositional data of class data.frame or matrix
- `alpha`: significance level
- `R`: Number of Monte Carlo simulations in order to provide p-values.

Summary

Selects Anderson-Darling tests (Anderson and Darling, 1952) and provides p-values for the goodness-of-fit test. The function is designed to work with compositional data of class data.frame or matrix.
adtestWrapper

robustEst logical
... additional parameters for print and summary passed through
object an object of class adtestWrapper for the summary method

Details

First, the data is transformed using the ‘ilr’-transformation. After applying this transformation
- all (D-1)-dimensional marginal, univariate distributions are tested using the univariate Anderson-
Darling test for normality.
- all 0.5 (D-1)(D-2)-dimensional bivariate angle distributions are tested using the Anderson-Darling
angle test for normality.
- the (D-1)-dimensional radius distribution is tested using the Anderson-Darling radius test for nor-
mality.
A print and a summary method are implemented. The latter one provides a similar output is pro-
posed by (Pawlowsky-Glahn, et al. (2008). In addition to that, p-values are provided.

Value

res a list including each test result
check information about the rejection of the null hypothesis
alpha the underlying significance level
info further information which is used by the print and summary method.
est “standard” for standard estimation and “robust” for robust estimation

Author(s)

Matthias Templ and Karel Hron

References

Aitchison, J. (1986) The Statistical Analysis of Compositional Data Monographs on Statistics and

See Also

adtest, pivotCoord

Examples

data(machineOperators)
a <- adtestWrapper(machineOperators, R=50) # choose higher value of R
a
summary(a)
Description

Percentages of children, middle generation and elderly population in 195 countries.

Usage

data(ageCatWorld)

Format

A data frame with 195 rows and 4 variables

Details

• <15 Percentage of people with age below 15
• 15-60 Percentage of people with age between 15 and 60
• 60+ Percentage of people with age above 60
• country country of origin

The rows sum up to 100.

Author(s)

extracted by Karel Hron and Eva Fiserova, implemented by Matthias Templ

References


Examples

data(ageCatWorld)
str(ageCatWorld)
summary(ageCatWorld)
rowSums(ageCatWorld[, 1:3])
ternaryDiag(ageCatWorld[, 1:3])
plot(pivotCoord(ageCatWorld[, 1:3]))
alcohol

alcohol

alcohol consumptions by country and type of alcohol

Description

- country Country
- year Year
- beer Consumption of pure alcohol on beer (in percentages)
- wine Consumption of pure alcohol on wine (in percentages)
- spirits Consumption of pure alcohol on spirits (in percentages)
- other Consumption of pure alcohol on other beverages (in percentages)

Usage

data(alcohol)

Format

A data frame with 193 rows and 6 variables

Author(s)

Matthias Templ <matthias.templ@tuwien.ac.at>

Source

Transferred from the World Health Organisation website.

Examples

data("alcohol")
str(alcohol)
summary(alcohol)
alcoholreg

**Description**

- country Country
- year Year
- recorded Recorded alcohol consumption
- unrecorded Unrecorded alcohol consumption

**Usage**

```r
data(alcoholreg)
```

**Format**

A data frame with 6 rows and 4 variables

**Author(s)**

Matthias Templ <matthias.templ@tuwien.ac.at>

**Source**

Transferred from the World Health Organisation website.

**Examples**

```r
data("alcoholreg")
alcoholreg
```

arcticLake

**Description**

Sand, silt, clay compositions of 39 sediment samples at different water depths in an Arctic lake.

This data set can be found on page 359 of the Aitchison book (see reference).

**Usage**

```r
data(arcticLake)
```
balances

Format

A data frame with 39 rows and 3 variables

Details

- sand numeric vector of percentages of sand
- silt numeric vector of percentages of silt
- clay numeric vector of percentages of clay

The rows sum up to 100, except for rounding errors.

Author(s)

Mathias Templ <matthias.templ@tuwien.ac.at>

References


Examples

data(arcticLake)
str(arcticLake)
summary(arcticLake)
rowSums(arcticLake)
ternaryDiag(arcticLake)
plot(pivotCoord(arcticLake))

balances  Balance calculation

Description

Given a D-dimensional compositional data set and a sequential binary partition, the function bal calculates the balances in order to express the given data in the (D-1)-dimensional real space.

Usage

balances(x, y)

Arguments

x  data frame or matrix, typically compositional data

y  binary partition
Details

The sequential binary partition constructs an orthonormal basis in the (D-1)-dimensional hyperplane in real space, resulting in orthonormal coordinates with respect to the Aitchison geometry of compositional data.

Value

balances

The balances represent orthonormal coordinates which allow an interpretation in sense of groups of compositional parts. Output is a matrix, the D-1 columns contain balance coordinates of the observations in the rows.

V

A Dx(D-1) contrast matrix associated with the orthonormal basis, corresponding to the sequential binary partition (in clr coefficients).

Author(s)

Veronika Pintar, Karel Hron, Matthias Templ

References


Examples

data(expenditures, package = "robCompositions")
y1 <- data.frame(c(1,1,1,-1,-1),c(1,-1,-1,0,0),
c(0,+1,-1,0,0),c(0,0,0,+1,-1))
y2 <- data.frame(c(1,-1,1,-1,-1),c(1,0,-1,0,0),
c(-1,-1,-1,1),c(0,-1,0,1,0))
y3 <- data.frame(c(1,1,1,-1),c(-1,-1,+1,0),
c(0,-1,0,1,0))
y4 <- data.frame(c(1,1,1,-1),c(-1,0,0,-1,1),
c(-1,-1,+1,0,0),c(-1,1,0,0,0))
y5 <- data.frame(c(1,1,1,-1),c(-1,-1,+1,0,0),
c(0,0,-1,1),c(-1,1,0,0,0))
b1 <- balances(expenditures, y1)
b2 <- balances(expenditures, y5)
b1$balances
b2$balances

data(machineOperators)
sbp <- data.frame(c(1,1,-1,-1),c(-1,+1,0,0),
c(0,0,+1,-1))
balances(machineOperators, sbp)
**biomarker**

---

**Description**

The function for identification of biomarkers and outlier diagnostics as described in paper "Robust biomarker identification in a two-class problem based on pairwise log-ratios"

**Usage**

```r
biomarker(
  x,
  cut = qnorm(0.975, 0, 1),
  g1,
  g2,
  type = "tau",
  diag = TRUE,
  plot = FALSE,
  diag.plot = FALSE
)
```

```
# S3 method for class 'biomarker'
plot(x, cut = qnorm(0.975, 0, 1), type = "Vstar", ...)

# S3 method for class 'biomarker'
print(x, ...)

# S3 method for class 'biomarker'
summary(object, ...)
```

**Arguments**

- **x** data frame
- **cut** cut-off value, initially set as 0.975 quantile of standard normal distribution
- **g1** vector with locations of observations of group 1
- **g2** vector with locations of observations of group 2
- **type** type of estimation of the variation matrix. Possible values are "sd", "mad" and "tau", representing Standard deviation, Median absolute deviation and Tau estimator of scale
- **diag** logical, indicating whether outlier diagnostic should be computed
- **plot** logical, indicating whether Vstar values should be plotted
- **diag.plot** logical, indicating whether outlier diagnostic plot should be made
- **...** further arguments can be passed through
- **object** object of class biomarker
Details

Robust biomarker identification and outlier diagnostics

The method computes variation matrices separately with observations from both groups and also together with all observations. Then, V statistics is then computed and normalized. The variables, for which according $V^*$ values are bigger that the cut-off value are considered as biomarkers.

Value

The function returns object of type "biomarker". Functions print, plot and summary are available.

- **biom.ident**: List of $V$, $V^*$, biomarkers
- **V**: Values of $V$ statistics
- **Vstar**: Normalizes values of $V$ statistics ($V^*$ values)
- **biomarkers**: Logical value, indicating if certain variable was identified as biomarker
- **diag**: Outlier diagnostics (returned only if diag=TRUE)

Author(s)

Jan Walach

See Also

- plot.biomarker

Examples

```r
# Data simulation
set.seed(4523)
n <- 40; p <- 50
r <- runif(p, min = 1, max = 10)
conc <- runif(p, min = 0, max = 1)*5+matrix(1,p,1)*5
a <- conc*r
S <- rnorm(n,0,0.3) %*% t(rep(1,p))
B <- matrix(rnorm(n*p,0,0.8),n,p)
R <- rep(1,n) %*% t(r)
M <- matrix(rnorm(n*p,0,0.021),n,p)
# Fifth observation is an outlier
M[5,] <- M[5,]*3 + sample(c(0.5,-0.5),replace=TRUE,p)
C <- rep(1,n) %*% t(conc)
C[1:20,c(2,15,28,40)] <- C[1:20,c(2,15,28,40)]+matrix(1,20,4)*1.8
X <- (1-S)*(C*R+B)*exp(M)
# Biomarker identification
b <- biomarker(X, g1 = 1:20, g2 = 21:40, type = "tau")
```
**Description**

Provides robust compositional biplots.

**Usage**

```r
## S3 method for class 'factanal'
biplot(x, ...)
```

**Arguments**

- `x` object of class ‘factanal’
- `...` ...

**Details**

The robust compositional biplot according to Aitchison and Greenacre (2002), computed from resulting (robust) loadings and scores, is performed.

**Value**

The robust compositional biplot.

**Author(s)**

M. Templ, K. Hron

**References**


**See Also**

`pfa`

**Examples**

```r
data(expenditures)
res.rob <- pfa(expenditures, factors=2, scores = "regression")
biplot(res.rob)
```
biplot.pcaCoDa

Biplot method

Description

Provides robust compositional biplots.

Usage

## S3 method for class 'pcaCoDa'
biplot(x, y, ...)

Arguments

x object of class ‘pcaCoDa’
y ...
... arguments passed to plot methods

Details

The robust compositional biplot according to Aitchison and Greenacre (2002), computed from (robust) loadings and scores resulting from \texttt{pcaCoDa}, is performed.

Value

The robust compositional biplot.

Author(s)

M. Templ, K. Hron

References


See Also

\texttt{pcaCoDa, plot.pcaCoDa}
Examples

```r
data(coffee)
p1 <- pcaCoDa(coffee[, -1])
biplot(p1)

## with labels for the scores:
data(arcticLake)ownames(arcticLake) <- paste(sample(letters[1:26], nrow(arcticLake), replace=TRUE),
1:nrow(arcticLake), sep="")
pc <- pcaCoDa(arcticLake, method="classical")
biplot(pc, xlabs=rownames(arcticLake))
```

---

`bootnComp`  
*Bootstrap to find optimal number of components*

**Description**
Combined bootstrap and cross validation procedure to find optimal number of PLS components

**Usage**

```r
bootnComp(X, y, R = 99, plotting = FALSE)
```

**Arguments**

- `X`: predictors as a matrix
- `y`: response
- `R`: number of bootstrap replicates
- `plotting`: if TRUE, a diagnostic plot is drawn for each bootstrap replicate

**Details**
Heavily used internally in function `impRZilr`.

**Value**
Including other information in a list, the optimal number of components

**Author(s)**
Matthias Templ
See Also

impRZilr

Examples

```r
## we refer to impRZilr()
```

cancer | *hospital discharges on cancer and distribution of age*

Description

Hospital discharges of in-patients on neoplasms (cancer) per 100,000 inhabitants (year 2007) and population age structure.

Format

A data set on 24 compositions on 6 variables.

Details

- country
- year
- p1 percentage of population with age below 15
- p2 percentage of population with age between 15 and 60
- p3 percentage of population with age above 60
- discharges hospital discharges of in-patients on neoplasms (cancer) per 100.000 inhabitants

The response (discharges) is provided for the European Union countries (except Greece, Hungary and Malta) by Eurostat. As explanatory variables we use the age structure of the population in the same countries (year 2008). The age structure consists of three parts, age smaller than 15, age between 15 and 60 and age above 60 years, and they are expressed as percentages on the overall population in the countries. The data are provided by the United Nations Statistics Division.

Author(s)

conversion to R by Karel Hron and Matthias Templ <matthias.templ@tuwien.ac.at>

Source


References

Examples

data(cancer)
str(cancer)

cancerMN  malignant neoplasms cancer

Description

Two main types of malignant neoplasms cancer affecting colon and lung, respectively, in male and female populations. For this purpose population data (2012) from 35 OECD countries were collected.

Format

A data set on 35 compositional tables on 4 parts (row-wise sorted cells) and 5 variables.

Details

- country
- females-colon number of colon cancer cases in female population
- females-lung number of lung cancer cases in female population
- males-colon number of colon cancer cases in male population
- males-lung number of lung cancer cases in male population

The data are obtained from the OECD website.

Author(s)
conversion to R by Karel Hron and integration by Matthias Templ <matthias.templ@tuwien.ac.at>

Source

https://www.oecd.org

Examples

data(cancerMN)
head(cancerMN)
rowSums(cancerMN[, 2:5])
ced

Compositional error deviation

Description

Normalized Aitchison distance between two data sets

Usage

`ced(x, y, ni)`

Arguments

- `x`: matrix or data frame
- `y`: matrix or data frame of the same size as `x`
- `ni`: normalization parameter. See details below.

Details

This function has been mainly written for procedures that evaluate imputation or replacement of rounded zeros. The `ni` parameter can thus, e.g. be used for expressing the number of rounded zeros.

Value

the compositional error distance

Author(s)

Matthias Templ

References


See Also

`rdcm`

Examples

```r
data(expenditures)
x <- expenditures
x[1,3] <- NA
xi <- impKNNa(x)$xImp
ced(expenditures, xi, ni = sum(is.na(x)))
```
Description
The centred logratio (clr) coefficients map D-part compositional data from the simplex into a D-dimensional real space.

Usage
```r
cenLR(x, base = exp(1))
```

Arguments
- `x`: multivariate data, ideally of class data.frame or matrix
- `base`: a positive or complex number: the base with respect to which logarithms are computed. Defaults to `exp(1)`.

Details
Each composition is divided by the geometric mean of its parts before the logarithm is taken.

Value
the resulting clr coefficients, including
- `x.clr`: clr coefficients
- `gm`: the geometric means of the original compositional data.

Note
The resulting data set is singular by definition.

Author(s)
Matthias Templ

References

See Also
- `cenLRinv`, `addLR`, `pivotCoord`, `addLRinv`, `pivotCoordInv`
Examples

data(expenditures)
eclr <- cenLR(expenditures)
inveclr <- cenLRinv(eclr)
head(expenditures)
head(inveclr)
head(pivotCoordInv(eclr$x.clr))

cenLRinv  Inverse centred logratio mapping

Description

Applies the inverse centred logratio mapping.

Usage

cenLRinv(x, useClassInfo = TRUE)

Arguments

x an object of class “clr”, “data.frame” or “matrix”
useClassInfo if the object is of class “clr”, the useClassInfo is used to determine if the class information should be used. If yes, also absolute values may be preserved.

Value

the resulting compositional data set.

Author(s)

Matthias Templ

References


See Also

cenLR, addLR, pivotCoord, addLRinv, pivotCoordInv
Examples

```r
data(expenditures)
eclr <- cenLR(expenditures, 2)
inveclr <- cenLRinv(eclr)
head(expenditures)
head(inveclr)
head(cenLRinv(eclr$x.clr))
```

---

**chorizonDL**  
*C-horizon of the Kola data with rounded zeros*

### Description

This data set is almost the same as the 'chorizon' data set in package `mvoutlier` and `chorizonDL`, except that values below the detection limit are coded as zeros, and detection limits provided as attributes to the data set and less variables are included.

### Format

A data frame with 606 observations on the following 62 variables.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>a numeric vector</td>
</tr>
<tr>
<td>XCOO</td>
<td>a numeric vector</td>
</tr>
<tr>
<td>YCOO</td>
<td>a numeric vector</td>
</tr>
<tr>
<td>Ag</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Al</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Al_XRF</td>
<td>concentration in wt. percentage</td>
</tr>
<tr>
<td>As</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Ba</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Ba_INAA</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Be</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Bi</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Ca</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Ca_XRF</td>
<td>concentration in wt. percentage</td>
</tr>
<tr>
<td>Cd</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Ce_INAA</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Co</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Co_INAA</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Cr</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Cr_INAA</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Element</td>
<td>Concentration</td>
</tr>
<tr>
<td>---------</td>
<td>---------------</td>
</tr>
<tr>
<td>Cu</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Fe</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Fe_XRF</td>
<td>concentration in wt. percentage</td>
</tr>
<tr>
<td>Hf</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>K</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>K_XRF</td>
<td>concentration in wt. percentage</td>
</tr>
<tr>
<td>La</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Li</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Mg</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Mg_XRF</td>
<td>concentration in wt. percentage</td>
</tr>
<tr>
<td>Mn</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Mn_XRF</td>
<td>concentration in wt. percentage</td>
</tr>
<tr>
<td>Na</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Na_XRF</td>
<td>concentration in wt. percentage</td>
</tr>
<tr>
<td>Nd</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Ni</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>P</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>P_XRF</td>
<td>concentration in wt. percentage</td>
</tr>
<tr>
<td>Pb</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>S</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Sc</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Si</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Si_XRF</td>
<td>concentration in wt. percentage</td>
</tr>
<tr>
<td>Sm</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Sr</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Th</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Ti</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Ti_XRF</td>
<td>concentration in wt. percentage</td>
</tr>
<tr>
<td>V</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Y</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Yb</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>Zn</td>
<td>concentration in mg/kg</td>
</tr>
<tr>
<td>LOI</td>
<td>concentration in wt. percentage</td>
</tr>
</tbody>
</table>

chorizonDL
pH  ph value
ELEV elevation
*COUN country
*ASP  a numeric vector
TOPC  a numeric vector
LITO  information on lithography

Note
For a more detailed description of this data set, see ‘chorizon’ in package mvoutlier.

Source

References

See Also
‘chorizon’, chorizonDL

Examples

data(chorizonDL, package = "robCompositions")
dim(chorizonDL)
colnames(chorizonDL)
zeroPatterns(chorizonDL)

---

clustCoDa  
Cluster analysis for compositional data

Description
Clustering in orthonormal coordinates or by using the Aitchison distance
Usage

clustCoDa(
  x,
  k = NULL,
  method = "Mclust",
  scale = "robust",
  transformation = "pivotCoord",
  distMethod = NULL,
  iter.max = 100,
  vals = TRUE,
  alt = NULL,
  bic = NULL,
  verbose = TRUE
)

## S3 method for class 'clustCoDa'
plot(
  x,
  y,
  ...,
  normalized = FALSE,
  which.plot = "clusterMeans",
  measure = "silwidths"
)

Arguments

x compositional data represented as a data.frame
k number of clusters
method clustering method. One of Mclust, cmeans, kmeansHartigan, cmeansUfcl, pam, clara, fanny, ward.D2, single, hclustComplete, average, mcquitty, median, centroid
scale if orthonormal coordinates should be normalized.
transformation default are the isometric logratio coordinates. Can only used when distMethod is not Aitchison.
distMethod Distance measure to be used. If "Aitchison", then transformation should be "identity".
iter.max parameter if kmeans is chosen. The maximum number of iterations allowed
vals if cluster validity measures should be calculated
alt a known partitioning can be provided (for special cluster validity measures)
bic if TRUE then the BIC criteria is evaluated for each single cluster as validity measure
verbose if TRUE additional print output is provided
y the y coordinates of points in the plot, optional if x is an appropriate structure.
... additional parameters for print method passed through
normalized results gets normalized before plotting. Normalization is done by z-transformation applied on each variable.

which.plot currently the only plot. Plot of cluster centers.

measure cluster validity measure to be considered for which.plot equals “partMeans”

Details

The compositional data set is either internally represented by orthonormal coordinates before a cluster algorithm is applied, or - depending on the choice of parameters - the Aitchison distance is used.

Value

all relevant information such as cluster centers, cluster memberships, and cluster statistics.

Author(s)

Matthias Templ (accessing the basic features of hclust, Mclust, kmeans, etc. that are all written by others)

References


Examples

data(expenditures)
x <- expenditures
rr <- clustCoDa(x, k=6, scale = "robust", transformation = "pivotCoord")
rr2 <- clustCoDa(x, k=6, distMethod = "Aitchison", scale = "none",
transformation = "identity")
rr3 <- clustCoDa(x, k=6, distMethod = "Aitchison", method = "single",
transformation = "identity", scale = "none")

## Not run:
require(reshape2)
plot(rr)
plot(rr, normalized = TRUE)
plot(rr, normalized = TRUE, which.plot = "partMeans")

## End(Not run)
clustCoDa_qmode  

Q-mode cluster analysis for compositional parts

Description

Clustering using the variation matrix of compositional parts

Usage

clustCoDa_qmode(x, method = "ward.D2")

Arguments

x  
compositional data represented as a data.frame

method  
hclust method

Value

a hclust object

Author(s)

Matthias Templ (accessing the basic features of hclust that are all written by other authors)

References


Examples

data(expenditures)
x <- expenditures
c1 <- clustCoDa_qmode(x)
## Not run:
require(reshape2)
plot(c1)
c2 <- clustCoDa_qmode(x, method = "single")
plot(c2)
## End(Not run)
Description

30 commercially available coffee samples of different origins.

Usage

data(coffee)

Format

A data frame with 30 observations and 7 variables.

Details

- sort sort of coffee
- acit acetic acid
- metpyr methylpyrazine
- furfu furfural
- furfualc furfuryl alcohol
- dimeth 2,6 dimethylpyrazine
- met5 5-methylfurfural

In the original data set, 15 volatile compounds (descriptors of coffee aroma) were selected for a statistical analysis. We selected six compounds (compositional parts) on three sorts of coffee.

Author(s)

Matthias Templ <matthias.templ@tuwien.ac.at>, Karel Hron

References


Examples

data(coffee)
str(coffee)
summary(coffee)
compareMahal  

*Compares Mahalanobis distances from two approaches*

**Description**

Mahalanobis distances are calculated for each zero pattern. Two approaches are used. The first one estimates Mahalanobis distance for observations belonging to one each zero pattern each. The second method uses a more sophisticated approach described below.

**Usage**

```r
compareMahal(x, imp = "KNNa")
```

```r
## S3 method for class 'mahal'
plot(x, y, ...)
```

**Arguments**

- **x**  data frame or matrix
- **imp**  imputation method
- **y**  unused second argument for the plot method
- **...**  additional arguments for plotting passed through

**Value**

- **df**  a data frame containing the Mahalanobis distances from the estimation in subgroups, the Mahalanobis distances from the imputation and covariance approach, an indicator specifying outliers and an indicator specifying the zero pattern
- **df2**  a groupwise statistics.

**Author(s)**

Matthias Templ, Karel Hron

**References**


**See Also**

`impKNNa`, `pivotCoord`
compositionalSpline

Examples

data(arcticLake)
# generate some zeros
arcticLake[1:10, 1] <- 0
arcticLake[11:20, 2] <- 0
m <- compareMahal(arcticLake)
plot(m)

compositionalSpline  Compositional spline

Description

This code implements the compositional smoothing splines grounded on the theory of Bayes spaces.

Usage

compositionalSpline(
  t,
  clrf,
  knots,
  w,
  order,
  der,
  alpha,
  spline.plot = FALSE,
  basis.plot = FALSE
)

Arguments

t  class midpoints
clrf  clr transformed values at class midpoints, i.e., fcenLR(f(t))
knots  sequence of knots
w  weights
order  order of the spline (i.e., degree + 1)
der  lth derivation
alpha  smoothing parameter
spline.plot  if TRUE, the resulting spline is plotted
basis.plot  if TRUE, the ZB-spline basis system is plotted
Details
The compositional splines enable to construct a spline basis in the centred logratio (clr) space of density functions (ZB-spline basis) and consequently also in the original space of densities (CB-spline basis). The resulting compositional splines in the clr space as well as the ZB-spline basis satisfy the zero integral constraint. This enables to work with compositional splines consistently in the framework of the Bayes space methodology.

Augmented knot sequence is obtained from the original knots by adding #(order-1) multiple end-points.

Value

- \( J \) value of the functional J
- \( \text{ZB\_coef} \) ZB-spline basis coefficients
- \( \text{CV} \) score of cross-validation
- \( \text{GCV} \) score of generalized cross-validation

Author(s)
J. Machalova <jitka.machalova@upol.cz>, R. Talska <talskarenata@seznam.cz>

References

---

constSum | Constant sum

### Description
Closes compositions to sum up to a given constant (default 1), by dividing each part of a composition by its row sum.

### Usage
\[
\text{constSum}(x, \text{const} = 1, \text{na.rm} = \text{TRUE})
\]

### Arguments
- **x** multivariate data ideally of class data.frame or matrix
- **const** constant, the default equals 1.
- **na.rm** removing missing values.

### Value
The data for which the row sums are equal to const.
**coord**

**Author(s)**

Mathias Templ

**Examples**

```r
data(expenditures)
constSum(expenditures)
constSum(expenditures, 100)
```

---

**Description**

General approach to orthonormal coordinates for compositional tables

**Usage**

```r
coord(x, SBPr, SBPc)
```

```r
## S3 method for class 'coord'
print(x, ...)
```

**Arguments**

- `x`: an object of class “table”, “data.frame” or “matrix”
- `SBPr`: sequential binary partition for rows
- `SBPc`: sequential binary partition for columns
- `...`: further arguments passed to the print function

**Details**

A contingency or probability table can be considered as a two-factor composition, we refer to compositional tables. This function constructs orthonormal coordinates for compositional tables using the balances approach for given sequential binary partitions on rows and columns of the compositional table.

**Value**

Row and column balances and odds ratios as coordinate representations of the independence and interaction tables, respectively.

- `row_balances`: row balances
- `row_bin`: binary partition for rows
col_balances  column balances
col_bin  binary partition for columns
odds_ratios_coord  odds ratio coordinates

Author(s)
Kamila Facevicova, and minor adaption by Matthias Templ

References

Examples

```r
x <- rbind(c(1,5,3,6,8,4),c(6,4,9,5,8,12),c(15,2,68,42,11,6),
          c(20,15,4,6,23,8),c(11,20,35,26,44,8))
x
SBPc <- rbind(c(1,1,1,1,-1,-1),c(1,-1,-1,-1,0,0),c(0,1,1,-1,0,0),
              c(0,1,-1,0,0,0),c(0,0,0,0,1,-1))
SBPc
SBPr <- rbind(c(1,1,1,-1,-1),c(1,1,-1,0,0),c(1,-1,0,0,0),c(0,0,0,1,-1))
SBPr
result <- coord(x, SBPr,SBPc)
result
data(socExp)
```

corCoDa  Correlations for compositional data

Description
This function computes correlation coefficients between compositional parts based on symmetric pivot coordinates.

Usage
corCoDa(x, ...)

Arguments
x  a matrix or data frame with compositional data
...  additional arguments for the function cor

Value
A compositional correlation matrix.
cubeCoord

Author(s)
Petra Kynclova

References

Examples
```r
data(expenditures)
corCoDa(expenditures)
x <- arcticLake
corCoDa(x)
```

---

cubeCoord | Coordinate representation of a compositional cube and of a sample of compositional cubes

Description
cubeCoord computes a system of orthonormal coordinates of a compositional cube. Computation of either pivot coordinates or a coordinate system based on the given SBP is possible.

Wrapper (cubeCoordWrapper): For each compositional cube in the sample cubeCoordWrapper computes a system of orthonormal coordinates and provide a simple descriptive analysis. Computation of either pivot coordinates or a coordinate system based on the given SBP is possible.

Usage
cubeCoord(
  x,
  row.factor = NULL,
  col.factor = NULL,
  slice.factor = NULL,
  value = NULL,
  SBPr = NULL,
  SBPc = NULL,
  SBPs = NULL,
  pivot = FALSE,
  print.res = FALSE
)

cubeCoordWrapper(
  x,
  obs.ID = NULL,
  row.factor = NULL,
  col.factor = NULL,
  slice.factor = NULL,
  value = NULL,
  SBPr = NULL,
  SBPc = NULL,
  SBPs = NULL,
  pivot = FALSE,
  print.res = FALSE
)
col.factor = NULL,
slice.factor = NULL,
value = NULL,
SBPr = NULL,
SBPc = NULL,
SBPs = NULL,
pivot = FALSE,
test = FALSE,
n.boot = 1000
)

Arguments

x a data frame containing variables representing row, column and slice factors of the respective compositional cube and variable with the values of the composition.

row.factor name of the variable representing the row factor. Needs to be stated with the quotation marks.

col.factor name of the variable representing the column factor. Needs to be stated with the quotation marks.

slice.factor name of the variable representing the slice factor. Needs to be stated with the quotation marks.

value name of the variable representing the values of the composition. Needs to be stated with the quotation marks.

SBPr an \( I - 1 \times I \) array defining the sequential binary partition of the values of the row factor, where \( I \) is the number of the row factor levels. The values assigned in the given step to the + group are marked by 1, values from the - group by -1 and the rest by 0. If it is not provided, the pivot version of coordinates is constructed automatically.

SBPc an \( J - 1 \times J \) array defining the sequential binary partition of the values of the column factor, where \( J \) is the number of the column factor levels. The values assigned in the given step to the + group are marked by 1, values from the - group by -1 and the rest by 0. If it is not provided, the pivot version of coordinates is constructed automatically.

SBPs an \( K - 1 \times K \) array defining the sequential binary partition of the values of the slice factor, where \( K \) is the number of the slice factor levels. The values assigned in the given step to the + group are marked by 1, values from the - group by -1 and the rest by 0. If it is not provided, the pivot version of coordinates is constructed automatically.

pivot logical, default is FALSE. If TRUE, or one of the SBPs is not defined, its pivot version is used.

print.res logical, default is FALSE. If TRUE, the output is displayed in the Console.

X a data frame containing variables representing row, column and slice factors of the respective compositional cubes, variable with the values of the composition and variable distinguishing the observations.
cubeCoord

obs.ID name of the variable distinguishing the observations. Needs to be stated with the quotation marks.

test logical, default is FALSE. If TRUE, the bootstrap analysis of coordinates is provided.
n.boot number of bootstrap samples.

Details
cubeCoord
This transformation moves the IJK-part compositional cubes from the simplex into a (IJK-1)-dimensional real space isometrically with respect to its three-factorial nature.

Wrapper (cubeCoordWrapper): Each of n IJK-part compositional cubes from the sample is with respect to its three-factorial nature isometrically transformed from the simplex into a (IJK-1)-dimensional real space. Sample mean values and standard deviations are computed and using bootstrap an estimate of 95 % confidence interval is given.

Value
Coordinates an array of orthonormal coordinates.
Grap.rep graphical representation of the coordinates. Parts denoted by + form the groups in the numerator of the respective computational formula, parts - form the denominator and parts . are not involved in the given coordinate.
Row.balances an array of row balances.
Column.balances an array of column balances.
Slice.balances an array of slice balances.
Row.column.OR an array of row-column OR coordinates.
Row.slice.OR an array of row-slice OR coordinates.
Column.slice.OR an array of column-slice OR coordinates.
Row.col.slice.OR an array of coordinates describing the mutual interaction between all three factors.
Contrast.matrix contrast matrix.
Log.ratios an array of pure log-ratios between groups of parts without the normalizing constant.
Coda.cube cube form of the given composition.
Bootstrap array of sample means, standard deviations and bootstrap confidence intervals.
Cubes Cube form of the given compositions.

Author(s)
Kamila Facevicova
References


See Also

tabCoord tabCoordWrapper

Examples

###############
### Coordinate representation of a CoDa Cube
## Not run:
### example from Facevicova (2019)
data(employment2)
CZE <- employment2[which(employment2$Country == 'CZE'), ]

# pivot coordinates
cubeCoord(CZE, "Sex", 'Contract', "Age", 'Value')

# coordinates with given SBP
r <- t(c(1,-1))
c <- t(c(1,-1))
s <- rbind(c(1,-1,-1), c(0,1,-1))
cubeCoord(CZE, "Sex", 'Contract', "Age", 'Value', r,c,s)

## End(Not run)

###############
### Analysis of a sample of CoDa Cubes
## Not run:
### example from Facevicova (2019)
data(employment2)
### Compositional tables approach,
### analysis of the relative structure.
### An example from Facevicova (2019)

# pivot coordinates
cubeCoordWrapper(employment2, 'Country', 'Sex', 'Contract', 'Age', 'Value', test=TRUE)

# coordinates with given SBP (defined in the paper)

r <- t(c(1,-1))
c <- t(c(1,-1))
s <- rbind(c(1,-1,-1), c(0,1,-1))
res <- cubeCoordWrapper(employment2, 'Country', 'Sex', 'Contract', "Age", 'Value', r,c,s, test=TRUE)
### Classical approach,
### generalized linear mixed effect model.

library(lme4)
employment2$y <- round(employment2$Value*1000)
glmer(y~Sex*Age*Contract+(1|Country),data=employment2,family=poisson)

### other relations within cube (in the log-ratio form)
### e.g. ratio between women and man in the group FT, 15to24
### and ratio between age groups 15to24 and 55plus

# transformation matrix
T <- rbind(c(1,rep(0,5), -1, rep(0,5)), c(rep(c(1/4,0,-1/4), 4)))
T %*% t(res$Contrast.matrix) %*% res$Bootstrap[,1]

## End(Not run)

---

**daCoDa**

*Linear and quadratic discriminant analysis for compositional data.*

**Description**

Linear and quadratic discriminant analysis for compositional data using either robust or classical estimation.

**Usage**

daCoDa(x, grp, coda = TRUE, method = "classical", rule = "linear", ...)

**Arguments**

- **x**: a matrix or data frame containing the explanatory variables
- **grp**: grouping variable: a factor specifying the class for each observation.
- **coda**: TRUE, when the underlying data are compositions.
- **method**: "classical" or "robust"
- **rule**: a character, either "linear" (the default) or "quadratic".
- **...**: additional arguments for the functions passed through

**Details**

Compositional data are expressed in orthonormal (ilr) coordinates (if coda==TRUE). For linear discriminant analysis the functions LdaClassic (classical) and Linda (robust) from the package rrcov are used. Similarly, quadratic discriminant analysis uses the functions QdaClassic and QdaCov (robust) from the same package.

The classical linear and quadratic discriminant rules are invariant to ilr coordinates and clr coefficients. The robust rules are invariant to ilr transformations if affine equivariant robust estimators of location and covariance are taken.
Value

An S4 object of class LdaClassic, Linda, QdaClassic or QdaCov. See package rrcov for details.

Author(s)

Jutta Gamper

References


See Also

LdaClassic, Linda, QdaClassic, QdaCov

Examples

```r
## toy data (non-compositional)
require(MASS)
x1 <- mvrnorm(20, c(0,0,0), diag(3))
x2 <- mvrnorm(30, c(3,0,0), diag(3))
x3 <- mvrnorm(40, c(0,3,0), diag(3))
X <- rbind(x1, x2, x3)
grp <- c(rep(1,20), rep(2,30), rep(3,40))
clas1 <- daCoDa(X, grp, coda=FALSE, method = "classical", rule="linear")
summary(clas1)
## predict runs only with newest version of rrcov
## Not run:
predict(clas1)

## End(Not run)
# specify different prior probabilities
clas2 <- daCoDa(X, grp, coda=FALSE, prior=c(1/3, 1/3, 1/3))
summary(clas2)

## compositional data
data(coffee)
x <- coffee[coffee$sort!="robusta",2:7]
shape <- droplevels(coffee$sort[coffee$sort!="robusta"])
cof.cla <- daCoDa(x, group, method="classical", rule="quadratic")
cof.rob <- daCoDa(x, group, method="robust", rule="quadratic")
## predict runs only with newest version of rrcov
## Not run:
predict(cof.cla)@ct
predict(cof.rob)@ct

## End(Not run)
```
**daFisher**

_Discriminant analysis by Fisher Rule._

---

**Description**

Discriminant analysis by Fisher's rule using the logratio approach to compositional data.

**Usage**

```r
daFisher(x, grp, coda = TRUE, method = "classical", plotScore = FALSE, ...)
```

```r
## S3 method for class 'daFisher'
print(x, ...)
```

```r
## S3 method for class 'daFisher'
predict(object, ..., newdata)
```

```r
## S3 method for class 'daFisher'
summary(object, ...)
```

**Arguments**

- **x**
  - a matrix or data frame containing the explanatory variables (training set)
- **grp**
  - grouping variable: a factor specifying the class for each observation.
- **coda**
  - TRUE, when the underlying data are compositions.
- **method**
  - "classical" or "robust" estimation.
- **plotScore**
  - TRUE, if the scores should be plotted automatically.
- **...**
  - additional arguments for the print method passed through
- **object**
  - object of class “daFisher”
- **newdata**
  - new data in the appropriate form (CoDa, etc)

**Details**

The Fisher rule leads only to linear boundaries. However, this method allows for dimension reduction and thus for a better visualization of the separation boundaries. For the Fisher discriminant rule (Fisher, 1938; Rao, 1948) the assumption of normal distribution of the groups is not explicitly required, although the method loses its optimality in case of deviations from normality.

The classical Fisher discriminant rule is invariant to ilr coordinates and clr coefficients. The robust rule is invariant to ilr transformations if affine equivariant robust estimators of location and covariance are taken.

Robustification is done (method “robust”) by estimating the columnwise means and the covariance by the Minimum Covariance Estimator.
Value

an object of class “daFisher” including the following elements

- \( B \) Between variance of the groups
- \( W \) Within variance of the groups
- loadings loadings
- scores fisher scores
- mc table indicating misclassifications
- mcrate misclassification rate
- coda coda
- grp grouping
- grppred predicted groups
- xc xc
- meanj meanj
- cv cv
- pj pj
- meanov meanov
- fdiscr fdiscr

Author(s)

Peter Filzmoser, Matthias Templ.

References


See Also

Linda

Examples

```r
## toy data (non-compositional)
require(MASS)
x1 <- mvrnorm(20,c(0,0,0),diag(3))
x2 <- mvrnorm(30,c(3,0,0),diag(3))
x3 <- mvrnorm(40,c(0,3,0),diag(3))
X <- rbind(x1,x2,x3)
grp=c(rep(1,20),rep(2,30),rep(3,40))
```

```
#par(mfrow=c(1,2))
d1 <- daFisher(X, grp=grp, method="classical", coda=FALSE)
d2 <- daFisher(X, grp=grp, method="robust", coda=FALSE)
d2
summary(d2)
predict(d2, newdata = X)

## example with olive data:
## Not run:
data(olive, package = "RnavGraph")
# exclude zeros (alternatively impute them if
# the detection limit is known using impRZilr())
ind <- which(olive == 0, arr.ind = TRUE)[,1]
olives <- olive[-ind, ]
x <- olives[, 4:10]
grp <- olives$Region # 3 groups
res <- daFisher(x, grp)
res
summary(res)
res <- daFisher(x, grp, plotScore = TRUE)
res <- daFisher(x, grp, method = "robust")
res
summary(res)
predict(res, newdata = x)
res <- daFisher(x, grp, plotScore = TRUE, method = "robust")

# 9 regions
grp <- olives$Area
res <- daFisher(x, grp, plotScore = TRUE)
res
summary(res)
predict(res, newdata = x)

## End(Not run)
```

economy
economic indicators

Description
Household and government consumptions, gross capital formation and import and exports of goods and services.

Usage
data(economy)

Format
A data frame with 30 observations and 7 variables
Details

- country country name
- country2 country name, short version
- HHconsumption Household and NPISH final consumption expenditure
- GOVconsumption Final consumption expenditure of general government
- capital Gross capital formation
- exports Exports of goods and services
- imports Imports of goods and services

Author(s)

Peter Filzmoser, Matthias Templ <matthias.templ@tuwien.ac.at>

References


Examples

data(economy)
str(economy)

educFM  
education level of father (F) and mother (M)

Description

Education level of father (F) and mother (M) in percentages of low (l), medium (m), and high (h) of 31 countries in Europe.

Usage

data(educFM)

Format

A data frame with 31 observations and 8 variables
Details

• country community code
• $F.l$ percentage of females with low education level
• $F.m$ percentage of females with medium education level
• $F.h$ percentage of females with high education level
• $F.l$ percentage of males with low education level
• $F.m$ percentage of males with medium education level
• $F.h$ percentage of males with high education level

Author(s)

Peter Filzmoser, Matthias Templ

Source

from Eurostat, https://ec.europa.eu/eurostat/

Examples

data(educFM)
str(educFM)

Description

Comprehensive European Food Consumption Database

Format

A data frame with 87 observations on the following 22 variables.

• Country country name
• Pop.Class population class
• grains Grains and grain-based products
• vegetables Vegetables and vegetable products (including fungi)
• roots Starchy roots and tubers
• nuts Legumes, nuts and oilseeds
• fruit Fruit and fruit products
• meat Meat and meat products (including edible offal)
• fish Fish and other seafood (including amphibians, rept)
- milk Milk and dairy products
- eggs Eggs and egg products
- sugar Sugar and confectionary
- fat Animal and vegetable fats and oils
- juices Fruit and vegetable juice
- nonalcoholic Non-alcoholic beverages (excepting milk based beverages)
- alcoholic Alcoholic beverages
- water Drinking water (water without any additives)
- herbs Herbs, spices and condiments
- small_children_food Food for infants and small children
- special Products for special nutritional use
- composite Composite food (including frozen products)
- snacks Snacks, desserts, and other foods

Details
The Comprehensive Food Consumption Database is a source of information on food consumption across the European Union (EU). The food consumption are reported in grams per day (g/day).

Source
efsa

Examples

data(efsa)

---

election  election data

description
Results of a election in Germany 2013 in different federal states

Usage
data(election)

Format
A data frame with 16 observations and 8 variables
Details

Votes for the political parties in the elections (compositional variables), and their relation to the unemployment rate and the average monthly income (external non-compositional variables). Votes are for the Christian Democratic Union and Christian Social Union of Bavaria, also called The Union (CDU/CSU), Social Democratic Party (SDP), The Left (DIE LINKE), Alliance '90/The Greens (GRUNE), Free Democratic Party (FDP) and the rest of the parties participated in the elections (other parties). The votes are examined in absolute values (number of valid votes). The unemployment in the federal states is reported in percentages, and the average monthly income in Euros.

- **CDU_CSU** Christian Democratic Union and Christian Social Union of Bavaria, also called The Union
- **SDP** Social Democratic Party
- **GRUENE** Alliance '90/The Greens
- **FDP** Free Democratic Party
- **DIE_LINKE** The Left
- **other_parties** Votes for the rest of the parties participated in the elections
- **unemployment** Unemployment in the federal states in percentages
- **income** Average monthly income in Euros

Author(s)

Petra Klynclova, Matthias Templ

Source

German Federal Statistical Office

References

Eurostat, [https://ec.europa.eu/eurostat/data](https://ec.europa.eu/eurostat/data)

Examples

data(election)
str(election)
Description

Results the Austrian presidential election in October 2016.

Usage

data(electionATbp)

Format

A data frame with 2202 observations and 10 variables

Details

Votes for the candidates Hofer and Van der Bellen.

- GKZ Community code
- Name Name of the community
- Eligible eligible votes
- Votes_total total votes
- Votes_invalid invalid votes
- Votes_valid valid votes
- Hofer_total votes for Hofer
- Hofer_perc votes for Hofer in percentages
- VanderBellen_total votes for Van der Bellen
- VanderBellen_perc votes for Van der Bellen in percentages

Author(s)

Peter Filzmoser

Source

OpenData Austria, https://www.data.gv.at/

Examples

data(electionATbp)
str(electionATbp)
**Description**

employment in different countries by gender and status.

**Usage**

data(employment)

**Format**

A three-dimensional table

**Examples**

data(employment)
str(employment)
str(employment)
employment

---

**Description**

Employment in different countries by Sex, Age, Contract, Value

**Usage**

data(employment2)

**Format**

A data.frame with 504 rows and 5 columns.

**Details**

For each country in the sample, an estimated number of employees in the year 2015 was available, divided according to gender and age of employees and the type of the contract. The data form a sample of 42 cubes with two rows (gender), two columns (type of contract) and three slices (age), which allow for a deeper analysis of the overall employment structure, not just from the perspective of each factor separately, but also from the perspective of the relations/interactions between them. Thorough analysis of the sample is described in Facovicova (2019).

- Country
• Sex: gender, males (M) and females (F)
• Age: age class, young (category 15 - 24), middle-aged (25 - 54) and older (55+) employees
• Contract: factor, defining the type of contract, full-time (FT) and part-time (PT) contracts
• Value: Number of employees in the given category (in thousands)

Author(s)
Kamila Facevicova

Source
https://stats.oecd.org

References

Examples
data(employment2)
head(employment2)

employment_df

Description
• gender: factor
• status: factor, defining if part or full time work
• country: factor
• value: employment

Usage
data(employment_df)

Format
A data.frame with 132 rows and 4 columns.

Examples
data(employment_df)
head(employment_df)
Description
This data set from Aitchison (1986), p. 395, describes household expenditures (in former Hong Kong dollars) on five commodity groups.

Usage
data(expenditures)

Format
A data frame with 20 observations on the following 5 variables.

Details

- housing housing (including fuel and light)
- foodstuffs foodstuffs
- alcohol alcohol and tobacco
- other other goods (including clothing, footwear and durable goods)
- services services (including transport and vehicles)

This data set contains household expenditures on five commodity groups of 20 single men. The variables represent housing (including fuel and light), foodstuff, alcohol and tobacco, other goods (including clothing, footwear and durable goods) and services (including transport and vehicles). Thus they represent the ratios of the men’s income spent on the mentioned expenditures.

Author(s)
Matthias Templ <matthias.templ@tuwien.ac.at>, Karel Hron

References

Examples
data(expenditures)
## imputing a missing value in the data set using k-nearest neighbor imputation:
expenditures[1,3]
expenditures[1,3] <- NA
impKNNa(expenditures)$xImp[1,3]
Mean consumption expenditure of households at EU-level. The final consumption expenditure of households encompasses all domestic costs (by residents and non-residents) for individual needs.

A data frame with 27 observations on the following 12 variables.

- Food numeric vector
- Alcohol numeric vector
- Clothing numeric vector
- Housing numeric vector
- Furnishings numeric vector
- Health numeric vector
- Transport numeric vector
- Communications numeric vector
- Recreation numeric vector
- Education numeric vector
- Restaurants numeric vector
- Other numeric vector

Source

Eurostat

Examples

data(expendituresEU)
fcenLR transformation (functional)

Description
fcenLR[lambda] transformation: mapping from B^2(lambda) into L^2(lambda)

Usage
fcenLR(z, z_step, density)

Arguments
z grid of points defining the abscissa
z_step step of the grid of the abscissa
density grid evaluation of the lambda-density

Value
out grid evaluation of the lambda-density in L^2(lambda)

Author(s)
R. Talska<talskarenata@seznam.cz>, A. Menafoglio, K. Hron<karel.hron@upol.cz>, J. J. Egozcue, J. Palarea-Albaladejo

References

Examples
# Example (normal density)
t = seq(-4.7, 4.7, length = 1000)
t_step = diff(t[1:2])
mean = 0; sd = 1.5
f = dnorm(t, mean, sd)
f1 = f/trapzc(t_step,f)
f.fcenLR = fcenLR(t,t_step,f)
f.fcenLRinv = fcenLRinv(t.fine,t_step,f.fcenLR)
plot(t,f.fcenLR, type="l",las=1, ylab="fcenLR(density)", cex.lab=1.2,cex.axis=1.2, col="darkblue",lwd=2) abline(h=0, col="red")
Inverse of fcenLR transformations (functional)

Description

Inverse of fcenLR transformations

Usage

fcenLRinv(z, z_step, fcenLR, k = 1)

Arguments

z  grid of points defining the abscissa
z_step  step of the grid of the abscissa
fcenLR  grid evaluation of (i) fcenLR[lambda] transformed lambda-density, (ii) fcenLR[u] transformed P-density, (iii) fcenLR[P] transformed P-density
k  value of the integral of density; if k=1 it returns a unit-integral representation of density

Details

By default, it returns a unit-integral representation of density.

Value

out ... grid evaluation of (i) lambda-density in B2(lambda), (ii) P-density in unweighted B2(lambda), (iii) P-density in B2(P)

Author(s)

R. Talska<talkrenata@seznam.cz>, A. Menafoglio, K. Hron<karel.hron@upol.cz>, J. J. Egozcue, J. Palarea-Albaladejo

Examples

# Example (normal density)
t = seq(-4.7,4.7, length = 1000)
t_step = diff(t[1:2])
mean = 0; sd = 1.5
f = dnorm(t, mean, sd)
f1 = f/trapzc(t_step,f)
fcenLRp

Description
fcenLR[p] transformation: mapping from B2(P) into L2(P)

Usage
fcenLRp(z, z_step, density, p)

Arguments
z        grid of points defining the abscissa
z_step   step of the grid of the abscissa
density  grid evaluation of the P-density
p        density of the reference measure P

Value
out      grid evaluation of the P-density in L2(P)

Author(s)
R. Talska <talskarenata@seznam.cz>, A. Menafoglio, K. Hron <karel.hron@upol.cz>, J.J. Egozcue, J. Palarea-Albaladejo

References
fcenLRu

fcenLRu transformation (functional)

Description
fcenLR[u] transformation: mapping from B2(P) into unweighted L2(lambda)

Usage
fcenLRu(z, z_step, density, p)

Arguments
z
grid of points defining the abscissa
z_step
step of the grid of the abscissa
density
grid evaluation of the P-density
p
density of the reference measure P

Value
out
grid evaluation of the P-density in unweighted L2(lambda)

Author(s)
R. Talska<talskarenata@seznam.cz>, A. Menafoglio, K. Hron<karel.hron@upol.cz>, J. J. Egozcue, J. Palarea-Albaladejo

References

Examples
# Common example for all transformations - fcenLR, fcenLRp, fcenLRu
# Example (log normal distribution under the reference P)
t = seq(1,10, length = 1000)
t_step = diff(t[1:2])

# Log normal density w.r.t. Lebesgue reference measure in B2(lambda)
f = dlnorm(t, meanlog = 1.5, sdlog = 0.5)

# Log normal density w.r.t. Lebesgue reference measure in L2(lambda)
f_fcenLR = fcenLR(t, t_step, f)

# New reference given by exponential density
p = dexp(t,0.25)/trapzc(t_step,dexp(t,0.25))
# Plot of log normal density w.r.t. Lebesgue reference measure
# in B2(\lambda) together with the new reference density p
matplot(t,f,type="l",las=1, ylab="density",cex.lab=1.2,cex.axis=1.2,
col="black",lwd=2,ylim=c(0,0.3),xlab="t")
matlines(t,p,col="blue")
text(2,0.25,”p”,col="blue")
text(4,0.22,”f”,col="black")

# Log-normal density w.r.t. exponential distribution in B2(\lambda)
# (unit-integral representation)
fp = (f/p)/trapzc(t_step,f/p)

# Log-normal density w.r.t. exponential distribution in L2(\lambda)
fp.fcenLRp = fcenLRp(t,t_step,fp,p)

# Log-normal density w.r.t. exponential distribution in L2(\lambda)
fp.fcenLRu = fcenLRu(t,t_step,fp,p)

# Log-normal density w.r.t. exponential distribution in B2(\lambda)
fp.u = fcenLRinv(t,t_step,fp.fcenLRu)

# Plot
layout(rbind(c(1,2,3,4),c(7,8,5,6)))
par(cex=1.1)
plot(t, f.fcenLR, type=’l’, ylab=expression(fcenLR[\lambda](f)),
     xlab=’t’,las=1,ylim=c(-3,3),
main=expression(bold(atop(paste(’(a) Representation of f in ’, L^2, (\lambda)),’[not weighted]’))))
abline(h=0,col="red")

plot(t, f, type=’l’, ylab=expression(f[\lambda]),
     xlab=’t’,las=1,ylim=c(0,0.4),
main=expression(bold(atop(paste(’(b) Density f in ’, B^2, (\lambda)),’[not weighted]’))))

plot(t, fp, type=’l’, ylab=expression(f[P]), xlab=’t’,
     las=1,ylim=c(0,0.4),
main=expression(bold(atop(paste(’(c) Density f in ’, B^2, (P)),’[weighted with P]’))))

plot(t, fp.fcenLRp, type=’l’, ylab=expression(fcenLR[P](f[P])),
     xlab=’t’,las=1,ylim=c(-3,3),
main=expression(bold(atop(paste(’(d) Representation of f in ’, L^2, (P)),’[weighted with P]’))))
abline(h=0,col="red")

plot(t, fp.u, type=’l’, ylab=expression(paste(omega^(-1),(f[P]))),
     xlab=’t’,las=1,ylim=c(0,0.4),
main=expression(bold(atop(paste(’(e) Representation of f in ’, B^2, (\lambda)),’[unweighted]’))))

plot(t, fp.fcenLRu, type=’l’, ylab=expression(paste(fcenLR[u](f[P]))),
     xlab=’t’,las=1,ylim=c(-3,3),
main=expression(bold(atop(paste(’(f) Representation of f in ’, L^2, (\lambda)),’[unweighted]’))))
abline(h=0,col="red")
foodbalance  

Description

Food balance in each country (2018)

Format

A data frame with 115 observations on the following 116 variables.

• country
• Cereals -Excluding BeerFood balance on cereals
• ....... #'
• Alcohol -Non-FoodFood balance on alcohol

Source


Examples

data(foodbalance)

GDPsatis  

Description

Satisfaction of GDP in 31 countries. The GDP is measured per capita from the year 2012.

Usage

data(GDPsatis)

Format

A data frame with 31 observations and 8 variables
Details

- country community code
- gdp GDP per capita in 2012
- very.bad satisfaction very bad
- bad satisfaction bad
- moderately.bad satisfaction moderately bad
- moderately.good satisfaction moderately good
- good satisfaction good
- very.good satisfaction very good

Author(s)

Peter Filzmoser, Matthias Templ

Source

from Eurostat, https://ec.europa.eu/eurostat/

Examples

```r
data(GDPsatis)
str(GDPsatis)
```

Description

Geochemical data set on agricultural and grazing land soil

Usage

```r
data(gemas)
```

Format

A data frame with 2108 observations and 30 variables
Details

- COUNTRY country name
- longitude longitude in WGS84
- latitude latitude in WGS84
- Xcoord UTM zone east
- Ycoord UTM zone north
- MeanTemp Annual mean temperature
- AnnPrec Annual mean precipitation
- soilclass soil class
- sand sand
- silt silt
- clay clay
- Al Concentration of aluminum (in mg/kg)
- Ba Concentration of barium (in mg/kg)
- Ca Concentration of calzium (in mg/kg)
- Cr Concentration of chromium (in mg/kg)
- Fe Concentration of iron (in mg/kg)
- K Concentration of pottasium (in mg/kg)
- Mg Concentration of magnesium (in mg/kg)
- Mn Concentration of manganese (in mg/kg)
- Na Concentration of sodium (in mg/kg)
- Nb Concentration of niobium (in mg/kg)
- Ni Concentration of nickel (in mg/kg)
- P Concentration of phosphorus (in mg/kg)
- Si Concentration of silicium (in mg/kg)
- Sr Concentration of strontium (in mg/kg)
- Ti Concentration of titanium (in mg/kg)
- V Concentration of vanadium (in mg/kg)
- Y Concentration of yttrium (in mg/kg)
- Zn Concentration of zinc (in mg/kg)
- Zr Concentration of zirconium (in mg/kg)
- LOI Loss on ignition (in wt-percent)

The sampling, at a density of 1 site/2500 sq. km, was completed at the beginning of 2009 by collecting 2211 samples of agricultural soil (Ap-horizon, 0-20 cm, regularly ploughed fields), and 2118 samples from land under permanent grass cover (grazing land soil, 0-10 cm), according to an agreed field protocol. All GEMAS project samples were shipped to Slovakia for sample preparation, where they were air dried, sieved to <2 mm using a nylon screen, homogenised and split to subsamples for analysis. They were analysed for a large number of chemical elements. In this sample, the main elements by X-ray fluorescence are included as well as the composition on sand, silt, clay.
Author(s)

GEMAS is a cooperation project between the EuroGeoSurveys Geochemistry Expert Group and Eurometaux. Integration in R, Peter Filzmoser and Matthias Templ.

References


Examples

data(gemas)
str(gemas)

## sample sites
## Not run:
## Not run:
require(ggmap)
map <- get_map("europe", source = "stamen", maptype = "watercolor", zoom=4)
ggmap(map) + geom_point(aes(x=longitude, y=latitude), data=gemas)
map <- get_map("europe", zoom=4)
ggmap(map) + geom_point(aes(x=longitude, y=latitude), data=gemas, size=0.8)
## End(Not run)

Description

Gjovik geochemical data set

Format

A data frame with 615 observations and 63 variables.

- ID a numeric vector
- MAT type of material
- mE32wgs longitude
- mN32wgs latitude
- XCOO X coordinates
- YCOO Y coordinates
- ALT altitude
• kmNS some distance north-south
• kmSN some distance south-north
• LITHO lithologies
• Ag a numeric vector
• Al a numeric vector
• As a numeric vector
• Au a numeric vector
• B a numeric vector
• Ba a numeric vector
• Be a numeric vector
• Bi a numeric vector
• Ca a numeric vector
• Cd a numeric vector
• Ce a numeric vector
• Co a numeric vector
• Cr a numeric vector
• Cs a numeric vector
• Cu a numeric vector
• Fe a numeric vector
• Ga a numeric vector
• Ge a numeric vector
• Hf a numeric vector
• Hg a numeric vector
• In a numeric vector
• K a numeric vector
• La a numeric vector
• Li a numeric vector
• Mg a numeric vector
• Mn a numeric vector
• Mo a numeric vector
• Na a numeric vector
• Nb a numeric vector
• Ni a numeric vector
• P a numeric vector
• Pb a numeric vector
• Pd a numeric vector
• Pt a numeric vector
Gjovik

- Rb a numeric vector
- Re a numeric vector
- S a numeric vector
- Sb a numeric vector
- Sc a numeric vector
- Se a numeric vector
- Sn a numeric vector
- Sr a numeric vector
- Ta a numeric vector
- Te a numeric vector
- Th a numeric vector
- Tl a numeric vector
- U a numeric vector
- V a numeric vector
- W a numeric vector
- Y a numeric vector
- Zn a numeric vector
- Zr a numeric vector

Details

Geochemical data set. 41 sample sites have been investigated. At each site, 15 different sample materials have been collected and analyzed for the concentration of more than 40 chemical elements. Soil: CHO - C horizon, OHO - O horizon. Mushroom: LAC - milkcap. Plant: BIL - birch leaves, BLE - blueberry leaves, BLU - blueberry twigs, BTW - birch twigs, CLE - cowberry leaves, COW - cowberry twigs, EQU - horsetail, FER - fern, HYL - terrestrial moss, PIB - pine bark, SNE - spruce needles, SPR - spruce twigs.

Author(s)

Peter Filzmoser, Dominika Miksova

References


Examples

data(gjovik)
str(gjovik)
Description
This function calculates the geometric mean.

Usage
\texttt{gm(x)}

Arguments
\texttt{x} a vector

Details
\texttt{gm} calculates the geometric mean for all positive entries of a vector. Please note that there is a faster version available implemented with Rcpp but it currently do not pass CRAN checks cause of use of Rcpp11 features. This C++ version accounts for over- and underflows. It is placed in inst/doc

Author(s)
Matthias Templ

Examples
\texttt{gm(c(3,5,3,6,7))}

---

Description
Computes the geometric mean(s) of a numeric vector, matrix or data.frame

Usage
\texttt{gmean_sum(x, margin = NULL)}

Arguments
\texttt{x} matrix or data.frame with numeric entries
\texttt{margin} a vector giving the subscripts which the function will be applied over, 1 indicates rows, 2 indicates columns, 3 indicates all values.
Details

gmean_sum calculates the totals based on geometric means while gmean calculates geometric means on rows (margin = 1), on columns (margin = 2), or on all values (margin = 3)

Value

geometric means (if gmean is used) or totals (if gmean_sum is used)

Author(s)

Matthias Templ

Examples

data("precipitation")
gmean_sum(precipitation)
gmean_sum(precipitation, margin = 2)
gmean_sum(precipitation, margin = 1)
gmean_sum(precipitation, margin = 3)
addmargins(precipitation)
addmargins(precipitation, FUN = gmean_sum)
addmargins(precipitation, FUN = mean)
addmargins(precipitation, FUN = gmean)

data("arcticLake", package = "robCompositions")
gmean(arcticLake$sand)
gmean(as.numeric(arcticLake[1, ]))
gmean(arcticLake)
gmean(arcticLake, margin = 1)
gmean(arcticLake, margin = 2)
gmean(arcticLake, margin = 3)

govexp

<table>
<thead>
<tr>
<th>govexp</th>
<th>government spending</th>
</tr>
</thead>
</table>

Description

Government expenditures based on COFOG categories

Format

A (tidy) data frame with 5140 observations on the following 4 variables.

- country Country of origin
- category The COFOG expenditures are divided into the following ten categories: general public services; defence; public order and safety; economic affairs; environmental protection; housing and community amenities; health; recreation, culture and religion; education; and social protection.
- year Year
- value COFOG spendings/expenditures
Details
The general government sector consists of central, state and local governments, and the social security funds controlled by these units. The data are based on the system of national accounts, a set of internationally agreed concepts, definitions, classifications and rules for national accounting. The classification of functions of government (COFOG) is used as classification system. The central government spending by category is measured as a percentage of total expenditures.

Author(s)
translated from https://data.oecd.org/ and restructured by Matthias Templ

Source
OECD: https://data.oecd.org/

Examples

data(govexp)
str(govexp)

Description
Distribution of European Y-chromosome DNA (Y-DNA) haplogroups by region in percentage.

Format
A data frame with 38 observations on the following 12 variables.

- I1 pre-Germanic (Nordic)
- I2b pre-Celto-Germanic
- I2a1 Sardinian, Basque
- I2a2 Dinaric, Danubian
- N1c1 Uralo-Finnic, Baltic, Siberian
- R1a Balto-Slavic, Mycenaean Greek, Macedonia
- R1b Italic, Celtic, Germanic; Hitite, Armenian
- G2a Caucasian, Greco-Anatolien
- E1b1b North and Eastern Afrika, Near Eastern, Balkanic
- J2 Mesopotamian, Minoan Greek, Phoenician
- J1 Semitic (Arabic, Jewish)
- T Near-Eastern, Egyptian, Ethiopian, Arabic
Details

Human Y-chromosome DNA can be divided in genealogical groups sharing a common ancestor, called haplogroups.

Source

Eupedia: [https://www.eupedia.com/europe/european_y-dna_haplogroups.shtml](https://www.eupedia.com/europe/european_y-dna_haplogroups.shtml)

Examples

```r
data(haplogroups)
```

<table>
<thead>
<tr>
<th>honey</th>
<th>honey compositions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Description

The contents of honey, syrup, and adulteration mineral elements.

Format

A data frame with 429 observations on the following 17 variables.

- `class` adulterated honey, Honey or Syrup
- `group` group information
- `group3` detailed group information
- `group1` less detailed group information
- `region` region
- `Al` chemical element
- `B` chemical element
- `Ba` chemical element
- `Ca` chemical element
- `Fe` chemical element
- `K` chemical element
- `Mg` chemical element
- `Mn` chemical element
- `Na` chemical element
- `P` chemical element
- `Sr` chemical element
- `Zn` chemical element
Details

Discrimination of honey and adulteration by elemental chemometrics profiling.

Note

In the original paper, sparse PLS-DA were applied to optimize the classify model and test effectiveness. Classify accuracy were exceed 87.7 percent.

Source

Mendeley Data, contributed by Liping Luo and translated to R by Matthias Templ

References


Examples

data(honey)

ilr.2x2 ilr coordinates in 2x2 compositional tables

Description

ilr coordinates of original, independent and interaction compositional table using SBP1 and SBP2

Usage

ilr.2x2(x, margin = 1, type = "independence", version = "book")

Arguments

x a 2x2 table
margin for 2x2 tables available for a whole set of another dimension. For example, if 2x2 tables are available for every country.
type choose between “independence” or “interaction” table
version the version used in the “paper” below or the version of the “book”.

Value

The ilr coordinates
Author(s)

Kamila Facevicova, Matthias Templ

References


Examples

data(employment)
ilr.2x2(employment[,,'AUT'])
ilr.2x2(employment[,,'AUT'], version = "paper")
ilr.2x2(employment, margin = 3, version = "paper")
ilr.2x2(employment[,,'AUT'], type = "interaction")

---

impAll

*Replacement of rounded zeros and missing values.*

Description

Parametric replacement of rounded zeros and missing values for compositional data using classical and robust methods based on ilr coordinates with special choice of balances. Values under detection limit should be saved with the negative value of the detection limit (per variable). Missing values should be coded as NA.

Usage

impAll(x)

Arguments

x data frame

Details

This is a wrapper function that calls `impRZilr()` for the replacement of zeros and `impCoda` for the imputation of missing values sequentially. The detection limit is automatically derived form negative numbers in the data set.

Value

The imputed data set.

Note

This function is mainly used by the compositionsGUI.
ImpCoda

References


See Also

impCoda, impRZilr

Examples

```r
## see the compositionsGUI
```

---

**impCoda**

*Imputation of missing values in compositional data*

**Description**

This function offers different methods for the imputation of missing values in compositional data. Missing values are initialized with proper values. Then iterative algorithms try to find better estimations for the former missing values.

**Usage**

```r
impCoda(  
  x,  
  maxit = 10,  
  eps = 0.5,  
  method = "ltsReg",  
  closed = FALSE,  
  init = "KNN",  
  k = 5,  
  dl = rep(0.05, ncol(x)),  
  noise = 0.1,  
  bruteforce = FALSE  
)
```
Arguments

- **x**: data frame or matrix
- **maxit**: maximum number of iterations
- **eps**: convergence criteria
- **method**: imputation method
- **closed**: imputation of transformed data (using ilr transformation) or in the original space (closed equals TRUE)
- **init**: method for initializing missing values
- **k**: number of nearest neighbors (if init $==$ “KNN”)
- **dl**: detection limit(s), only important for the imputation of rounded zeros
- **noise**: amount of adding random noise to predictors after convergency
- **bruteforce**: if TRUE, imputations over dl are set to dl. If FALSE, truncated (Tobit) regression is applied.

Details

eps: The algorithm is finished as soon as the imputed values stabilize, i.e. until the sum of Aitchison distances from the present and previous iteration changes only marginally (eps).

method: Several different methods can be chosen, such as ‘ltsReg’: least trimmed squares regression is used within the iterative procedure. ‘lm’: least squares regression is used within the iterative procedure. ‘classical’: principal component analysis is used within the iterative procedure. ‘ltsReg2’: least trimmed squares regression is used within the iterative procedure. The imputed values are perturbed in the direction of the predictor by values drawn form a normal distribution with mean and standard deviation related to the corresponding residuals and multiplied by noise.

Value

- **xOrig**: Original data frame or matrix
- **xImp**: Imputed data
- **criteria**: Sum of the Aitchison distances from the present and previous iteration
- **iter**: Number of iterations
- **maxit**: Maximum number of iterations
- **w**: Amount of imputed values
- **wind**: Index of the missing values in the data

Author(s)

Matthias Templ, Karel Hron

References

impKNNa

Imputation of missing values in compositional data using knn methods

Description

This function offers several k-nearest neighbor methods for the imputation of missing values in compositional data.

Usage

impKNNa(  
x,  
method = "knn",  
k = 3,  
metric = "Aitchison",  
agg = "median",  
primitive = FALSE,  
normknn = TRUE,  
das = FALSE,  
adj = "median"  
)

Arguments

x data frame or matrix

method method (at the moment, only “knn” can be used)

k number of nearest neighbors chosen for imputation

Examples

data(expenditures)  
x <- expenditures  
x[1,3]  
x[1,3] <- NA  
xi <- impCoda(x)$xImp  
xi[1,3]  
s1 <- sum(x[1,-3])  
imps <- sum(xi[1,-3])  
xi[,3] * s1/imps

# other methods  
impCoda(x, method = "lm")  
impCoda(x, method = "ltsReg")

See Also

impKNNa, pivotCoord
**impKNNa**

- **metric**: “Aichison” or “Euclidean”
- **agg**: “median” or “mean”, for the aggregation of the nearest neighbors
- **primitive**: if TRUE, a more enhanced search for the $k$-nearest neighbors is obtained (see details)
- **normknn**: An adjustment of the imputed values is performed if TRUE
- **das**: deprecated. if TRUE, the definition of the Aitchison distance, based on simple logratios of the compositional part, is used (Aitchison, 2000) to calculate distances between observations. if FALSE, a version using the clr transformation is used.
- **adj**: either ‘median’ (default) or ‘sum’ can be chosen for the adjustment of the nearest neighbors, see Hron et al., 2010.

**Details**

The Aitchison metric should be chosen when dealing with compositional data, the Euclidean metric otherwise.

If `primitive` == FALSE, a sequential search for the $k$-nearest neighbors is applied for every missing value where all information corresponding to the non-missing cells plus the information in the variable to be imputed plus some additional information is available. If `primitive` == TRUE, a search of the $k$-nearest neighbors among observations is applied where in addition to the variable to be imputed any further cells are non-missing.

If `normknn` is TRUE (preferred option) the imputed cells from a nearest neighbor method are adjusted with special adjustment factors (more details can be found online (see the references)).

**Value**

- **xOrig**: Original data frame or matrix
- **xImp**: Imputed data
- **w**: Amount of imputed values
- **wind**: Index of the missing values in the data
- **metric**: Metric used

**Author(s)**

Matthias Templ

**References**


**See Also**

impCoda
impRZalr

**Examples**

```r
data(expenditures)
x <- expenditures
x[1, 3] <- NA
xi <- impKNNa(x)$xImp
xi[1, 3]
```

---

**impRZalr**

*alr EM-based imputation of rounded zeros*

**Description**

A modified EM alr-algorithm for replacing rounded zeros in compositional data sets.

**Usage**

```r
impRZalr(
  x,
  pos = ncol(x),
  dl = rep(0.05, ncol(x) - 1),
  eps = 1e-04,
  maxit = 50,
  bruteforce = FALSE,
  method = "lm",
  step = FALSE,
  nComp = "boot",
  R = 10,
  verbose = FALSE
)
```

**Arguments**

- `x`: compositional data
- `pos`: position of the rationing variable for alr transformation
- `dl`: detection limit for each part
- `eps`: convergence criteria
- `maxit`: maximum number of iterations
- `bruteforce`: if TRUE, imputations over dl are set to dl. If FALSE, truncated (Tobit) regression is applied.
- `method`: either “lm” (default) or “MM”
- `step`: if TRUE, a stepwise (AIC) procedure is applied when fitting models
impRZalr

nComp if determined, it fixes the number of pls components. If “boot”, the number of pls components are estimated using a bootstrapped cross validation approach.

R number of bootstrap samples for the determination of pls components. Only important for method “pls”.

verbose additional print output during calculations.

Details

Statistical analysis of compositional data including zeros runs into problems, because log-ratios cannot be applied. Usually, rounded zeros are considerer as missing not at random missing values. The algorithm first applies an additive log-ratio transformation to the compositions. Then the rounded zeros are imputed using a modified EM algorithm.

Value

xOrig Original data frame or matrix
xImp Imputed data
wind Index of the missing values in the data
iter Number of iterations
eps eps

Author(s)

Matthias Templ and Karel Hron

References


See Also

impRZilr

Examples

data(arcticLake)
x <- arcticLake
## generate rounded zeros artificially:
x[x[,1] < 5, 1] <- 0
x[x[,2] < 47, 2] <- 0
xia <- impRZalr(x, pos=3, dl=c(5,47), eps=0.05)
xia$xImp
impRZilr

EM-based replacement of rounded zeros in compositional data

Description

Parametric replacement of rounded zeros for compositional data using classical and robust methods based on ilr coordinates with a special choice of balances.

Usage

impRZilr(
  x,
  maxit = 10,
  eps = 0.1,
  method = "pls",
  dl = rep(0.05, ncol(x)),
  variation = FALSE,
  nComp = "boot",
  bruteforce = FALSE,
  noisemethod = "residuals",
  noise = FALSE,
  R = 10,
  correction = "normal",
  verbose = FALSE
)

Arguments

x       data.frame or matrix
maxit   maximum number of iterations
eps     convergency criteria
method  either "lm", "MM" or "pls"
dl      Detection limit for each variable. zero for variables with variables that have no detection limit problems.
variation matrix is used to first select number of parts
nComp   if determined, it fixes the number of pls components. If "boot", the number of pls components are estimated using a bootstrapped cross validation approach.
bruteforce sets imputed values above the detection limit to the detection limit. Replacement above the detection limit only exceptionally occur due to numerical instabilities. The default is FALSE!
oisemethod adding noise to imputed values. Experimental
noise   TRUE to activate noise (experimental)
R       number of bootstrap samples for the determination of pls components. Only important for method "pls".
correction normal or density
verbose additional print output during calculations.
**Details**

Statistical analysis of compositional data including zeros runs into problems, because log-ratios cannot be applied. Usually, rounded zeros are considered as missing not at random missing values. The algorithm iteratively imputes parts with rounded zeros whereas in each step (1) compositional data are expressed in pivot coordinates (2) tobit regression is applied (3) the rounded zeros are replaced by the expected values (4) the corresponding inverse ilr mapping is applied. After all parts are imputed, the algorithm starts again until the imputations do not change.

**Value**

- **x**: imputed data
- **criteria**: change between last and second last iteration
- **iter**: number of iterations
- **maxit**: maximum number of iterations
- **wind**: index of zeros
- **nComp**: number of components for method pls
- **method**: chosen method

**Author(s)**

Matthias Templ and Peter Filzmoser

**References**


**See Also**

impRZalr

**Examples**

```r
data(arcticLake)
x <- arcticLake
## generate rounded zeros artificially:
#x[x[,1] < 5, 1] <- 0
x[x[,2] < 44, 2] <- 0
xia <- impRZilr(x, dl=c(5,44,0), eps=0.01, method="lm")
xia$x
```
imputeBDLs  

*EM-based replacement of rounded zeros in compositional data*

**Description**

Parametric replacement of rounded zeros for compositional data using classical and robust methods based on ilr coordinates with a special choice of balances.

**Usage**

```r
imputeBDLs(
  x,
  maxit = 10,
  eps = 0.1,
  method = "subPLS",
  dl = rep(0.05, ncol(x)),
  variation = TRUE,
  nPred = NULL,
  nComp = "boot",
  bruteforce = FALSE,
  noisemethod = "residuals",
  noise = FALSE,
  R = 10,
  correction = "normal",
  verbose = FALSE,
  test = FALSE
)
```

```r
adjustImputed(xImp, xOrig, wind)
```

```r
checkData(x, dl)
```

```r
## S3 method for class 'replaced'
print(x, ...)
```

**Arguments**

- `x`  
  data.frame or matrix
- `maxit`  
  maximum number of iterations
- `eps`  
  convergancy criteria
- `method`  
  either "lm", "lmrob" or "pls"
- `dl`  
  Detection limit for each variable. zero for variables with variables that have no detection limit problems.
- `variation`,  
  if TRUE those predictors are chosen in each step, who’s variation is lowest to the predictor.
imputeBDLs

n Pred, if determined and variation equals TRUE, it fixes the number of predictors
n Comp if determined, it fixes the number of pls components. If “boot”, the number of
pls components are estimated using a bootstraped cross validation approach.
bruteforce sets imputed values above the detection limit to the detection limit. Replacement
above the detection limit are only exceptionally occur due to numerical instabili-
ties. The default is FALSE!
noisemethod adding noise to imputed values. Experimental
noise TRUE to activate noise (experimental)
R number of bootstrap samples for the determination of pls components. Only
important for method “pls”.
correction normal or density
verbose additional print output during calculations.
test an internal test situation (this parameter will be deleted soon)
xImp imputed data set
xOrig original data set
wind index matrix of rounded zeros
... further arguments passed through the print function

Details

Statistical analysis of compositional data including zeros runs into problems, because log-ratios
cannot be applied. Usually, rounded zeros are considerer as missing not at random missing values.

The algorithm iteratively imputes parts with rounded zeros whereas in each step (1) compositional
data are expressed in pivot coordinates (2) tobit regression is applied (3) the rounded zeros are
replaced by the expected values (4) the corresponding inverse ilr mapping is applied. After all parts
are imputed, the algorithm starts again until the imputations do not change.

Value

x imputed data
criteria change between last and second last iteration
iter number of iterations
maxit maximum number of iterations
wind index of zeros
nComp number of components for method pls
method chosen method

Author(s)

Matthias Templ, method subPLS from Jiajia Chen
References

See Also
imputeBDLs

Examples

```r
# Generate random data
p <- 10
n <- 50
k <- 2
T <- matrix(rnorm(n*k), ncol=k)
B <- matrix(runif(p*k,-1,1),ncol=k)
X <- T %*% t(B)
E <- matrix(rnorm(n*p, 0,0.1), ncol=p)
XE <- X + E
data <- data.frame(pivotCoordInv(XE))
col <- ncol(data)
row <- nrow(data)
DL <- matrix(rep(0),ncol=col,nrow=1)
for(j in seq(1,col,2))
{DL[j] <- quantile(data[,j],probs=0.06,na.rm=FALSE)}
for(j in 1:col)
{data[data[,j]<DL[,j],j] <- 0}
## Not run:
# under dontrun because of long execution time
imp <- imputeBDLs(data,dl=DL,maxit=10,eps=0.1,R=10,method="subPLS")
imp
imp <- imputeBDLs(data,dl=DL,maxit=10,eps=0.1,R=10,method="pls", variation = FALSE)
imp
imp <- imputeBDLs(data,dl=DL,maxit=10,eps=0.1,R=10,method="lm")
imp
imp <- imputeBDLs(data,dl=DL,maxit=10,eps=0.1,R=10,method="lmrob")
imp
data(mcad)
## generate rounded zeros artificially:
x <- mcad
x <- x[1:25, 2:ncol(x)]
dl <- apply(x, 2, quantile, 0.1)
for(i in seq(1, ncol(x), 2)){
  x[x[,i] < dl[i], i] <- 0
}
ni <- sum(x==0, na.rm=TRUE)
```
imputeUDLs

Imputation of values above an upper detection limit in compositional data

Description

Parametric replacement of values above upper detection limit for compositional data using classical and robust methods (possibly also the pls method) based on ilr-transformations with special choice of balances.

Usage

```r
imputeUDLs(
x,  
maxit = 10,  
eps = 0.1,  
method = "lm",  
dl = NULL,  
variation = TRUE,  
nPred = NULL,  
nComp = "boot",  
bruteforce = FALSE,  
noisemethod = "residuals",  
noise = FALSE,  
R = 10,  
correction = "normal",  
verbose = FALSE
)
```

Arguments

- `x` data.frame or matrix
- `maxit` maximum number of iterations
imputeUDLs

eps convergency criteria
method either "lm", "lmrob" or "pls"
dl Detection limit for each variable. zero for variables with variables that have no detection limit problems.
variation, if TRUE those predictors are chosen in each step, who’s variation is lowest to the predictor.
nPred, if determined and variation equals TRUE, it fixes the number of predictors
nComp, if determined, it fixes the number of pls components. If “boot”, the number of pls components are estimated using a bootstraped cross validation approach.
bruteforce sets imputed values above the detection limit to the detection limit. Replacement above the detection limit are only exceptionly occur due to numerical instabilities. The default is FALSE!
noisemethod adding noise to imputed values. Experimental
noise TRUE to activate noise (experimental)
R number of bootstrap samples for the determination of pls components. Only important for method “pls”.
correction normal or density
verbose additional print output during calculations.

Details
imputeUDLs
An imputation method for right-censored compositional data. Statistical analysis is not possible with values reported in data, for example as ">10000". These values are replaced using tobit regression.

The algorithm iteratively imputes parts with values above upper detection limit whereas in each step (1) compositional data are expressed in pivot coordinates (2) tobit regression is applied (3) the values above upper detection limit are replaced by the expected values (4) the corresponding inverse ilr mapping is applied. After all parts are imputed, the algorithm starts again until the imputations only change marginally.

Value

x imputed data
criteria change between last and second last iteration
iter number of iterations
maxit maximum number of iterations
wind index of values above upper detection limit
nComp number of components for method pls
method chosen method

Author(s)

Peter Filzmoser, Dominika Miksova based on function imputeBDLs code from Matthias Templ
## References


## See Also

[imputeBDLs](#)

## Examples

```r
data(gemas) # read data
dat <- gemas[gemas$COUNTRY=="HEL",c(12:29)]
UDL <- apply(dat,2,max)
names(UDL) <- names(dat)
UDL["Mn"] <- quantile(dat[,"Mn"], probs = 0.8) # UDL present only in one variable
whichud1 <- dat[,"Mn"] > UDL["Mn"]

imp.lm <- dat
imp.lm[whichud1,"Mn"] <- Inf
res.lm <- imputeUDLs(imp.lm, dl=UDL, method="lm", variation=TRUE)
imp.lm <- res.lm$x
```

---

## ind2x2

### Independence 2x2 compositional table

#### Description

Estimates the expected frequencies from an 2x2 table under the null hypotheses of independence.

#### Usage

```r
ind2x2(x, margin = 3, pTabMethod = c("dirichlet", "half", "classical"))
```

#### Arguments

- **x**: a 2x2 table
- **margin**: if multidimensional table (larger than 2-dimensional), then the margin determines on which dimension the independene tables should be estimated.
- **pTabMethod**: ‘classical’ that is function `prop.table()` from package base or method “half” that add 1/2 to each cell to avoid zero problems.
indTab

Value

The independence table(s) with either relative or absolute frequencies.

Author(s)

Kamila Facevicova, Matthias Templ

References


Examples

data(employment)
ind2x2(employment)

indTab

Independence table

Description

Estimates the expected frequencies from an m-way table under the null hypotheses of independence.

Usage

\[
\text{indTab}(x, \text{margin} = \text{c("gmean_sum", "sum")}, \text{frequency} = \text{c("relative", "absolute")}, \text{pTabMethod} = \text{c("dirichlet", "half", "classical")})
\]

Arguments

\begin{itemize}
\item \text{x} an object of class table
\item \text{margin} determines how the margins of the table should be estimated (default via geometric mean margins)
\item \text{frequency} indicates whether absolute or relative frequencies should be computed.
\item \text{pTabMethod} to estimate the propability table. Default is ‘dirichlet’. Other available methods: ‘classical’ that is function \text{prop.table()} from package \text{base} or method ‘half’ that add 1/2 to each cell to avoid zero problems.
\end{itemize}

Details

Because of the compositional nature of probability tables, the independence tables should be estimated using geometric marginals.
Value

The independence table(s) with either relative or absolute frequencies.

Author(s)

Matthias Templ

References

tables using simplicial geometry. Communications in Statistics - Theory and Methods, 44 (18),
3978–3996.

Examples

data(precipitation)
tab1 <- indTab(precipitation)
tab1
sum(tab1)

## Not run:
data("PreSex", package = "vcd")
indTab(PreSex)

## End(Not run)

instw

value added, output and input for different ISIC codes and countries.

Description

• ctct
• isicISIC classification, Rev 3.2
• VAvalue added
• OUToutput
• INPinput
• IS03country code
• mhtmht

Usage

data(instw)

Format

A data.frame with 1555 rows and 7 columns.
Examples

data(instw)
head(instw)

## int2x2

### Interaction 2x2 table

Description
Estimates the interactions from an 2x2 table under the null hypotheses of independence.

Usage

```r
int2x2(x, margin = 3, pTabMethod = c("dirichlet", "half", "classical"))
```

Arguments

- `x` a 2x2 table
- `margin` if multidimensional table (larger than 2-dimensional), then the margin determines on which dimension the independene tables should be estimated.
- `pTabMethod` to estimate the probability table. Default is ‘dirichlet’. Other available methods: ‘classical’ that is function `prop.table()` from package base or method “half” that add 1/2 to each cell to avoid zero problems.

Value
The independence table(s) with either relative or absolute frequencies.

Author(s)
Kamila Facevicova, Matthias Templ

References


Examples

data(employment)
int2x2(employment)
intArray

**Description**

Estimates the interaction compositional table with normalization for further analysis according to Egozcue et al. (2015)

**Usage**

```r
intArray(x)
```

**Arguments**

- `x` an object of class “intTab”

**Details**

Estimates the interaction table using its ilr coordinates.

**Value**

The interaction array

**Author(s)**

Matthias Templ

**References**


**See Also**

`intTab`

**Examples**

```r
data(precipitation)
tab1prob <- prop.table(precipitation)
tab1 <- indTab(precipitation)
tabINT <- intTab(tab1prob, tab1)
intArray(tabINT)
```
intTab  

*Interaction table*

**Description**

Estimates the interaction table based on clr and inverse clr coefficients.

**Usage**

```r
intTab(x, y, frequencies = c("relative", "absolute"))
```

**Arguments**

- **x**: an object of class table
- **y**: the corresponding independence table which is of class “intTab”.
- **frequencies**: indicates whether absolute or relative frequencies should be computed.

**Details**

Because of the compositional nature of probability tables, the independence tables should be estimated using geometric marginals.

**Value**

- **intTab**: The interaction table(s) with either relative or absolute frequencies.
- **signs**: The sign illustrates if there is an excess of probability (plus), or a deficit (minus) regarding to the estimated probability table and the independence table in the clr space.

**Author(s)**

Matthias Templ

**References**


**Examples**

```r
data(precipitation)
tabloprob <- prop.table(precipitation)
tablo <- indTab(precipitation)
intTab(tablprob, tabl)
```
Description
Checks if two vectors or two data frames are from the same equivalence class

Usage
\[
is.equivalent(x, y, tolerance = \text{.Machine\$double.eps^{0.5}})
\]

Arguments
- \textit{x}: either a numeric vector, or a data.frame containing such vectors.
- \textit{y}: either a numeric vector, or a data.frame containing such vectors.
- \textit{tolerance}: numeric \texttt{\geq 0}. Differences smaller than tolerance are not considered.

Value
logical \texttt{TRUE} if the two vectors are from the same equivalence class.

Author(s)
Matthias Templ

References

See Also
\texttt{all.equal}

Examples
\begin{verbatim}
is.equivalent(1:10, 1:10*2)
is.equivalent(1:10, 1:10+1)
data(expenditures)
x <- expenditures
is.equivalent(x, constSum(x))
y <- x
y[1,1] <- x[1,1]+1
is.equivalent(y, constSum(x))
\end{verbatim}
isic32  ISIC codes by name

Description

• code ISIC code, Rev 3.2
• description Description of ISIC codes

Usage

data(isic32)

Format

A data.frame with 24 rows and 2 columns.

Examples

data(instw)
instw

laborForce  labour force by status in employment

Description

Labour force by status in employment for 124 countries, latest update: December 2009

Format

A data set on 124 compositions on 9 variables.

Details

• country country
• year year
• employeesW percentage female employees
• employeesM percentage male employees
• employersW percentage female employers
• employersM percentage male employers
• ownW percentage female own-account workers and contributing family workers
• ownM percentage male own-account workers and contributing family workers
• source HS: household or labour force survey. OE: official estimates. PC: population census
lifeExpGdp

**Author(s)**
conversion to R by Karel Hron and Matthias Templ <matthias.templ@tuwien.ac.at>

**Source**

**References**

**Examples**
```r
data(laborForce)
str(laborForce)
```

---

**lifeExpGdp**  
*life expectancy and GDP (2008) for EU-countries*

**Description**
Social-economic data for compositional regression.

**Format**
A data set on 27 compositions on 9 variables.

**Details**
- country
- agriculture GDP on agriculture, hunting, forestry, fishing (ISIC A-B, x1)
- manufacture GDP on mining, manufacturing, utilities (ISIC C-E, x2)
- construction GDP on construction (ISIC F, x3)
- wholesales GDP on wholesale, retail trade, restaurants and hotels (ISIC G-H, x4)
- transport GDP on transport, storage and communication (ISIC I, x5)
- other GDP on other activities (ISIC J-P, x6)
- lifeExpMen life expectancy for men and women
- lifeExpWomen life expectancy for men and women

**Author(s)**
conversion to R by Karel Hron and Matthias Templ <matthias.templ@tuwien.ac.at>
Source


References


Examples

```r
data(lifeExpGdp)
str(lifeExpGdp)
```

---

**lmCoDaX**

*Classical and robust regression of non-compositional (real) response on compositional predictors*

**Description**

Delivers appropriate inference for regression of y on a compositional matrix X.

**Usage**

```r
lmCoDaX(y, X, method = "robust")
```

**Arguments**

- **y** The response which should be non-compositional
- **X** The compositional predictors as a matrix, data.frame or numeric vector
- **method** If robust, LTS-regression is applied, while with method equals “classical”, the conventional least squares regression is applied.

**Details**

Compositional explanatory variables should not be directly used in a linear regression model because any inference statistic can become misleading. While various approaches for this problem were proposed, here an approach based on the pivot coordinates is used.

**Value**

An object of class ‘lts’ or ‘lm’ and two summary objects.

**Author(s)**

Peter Filzmoser
References


See Also

*lm*

Examples

```r
## How the total household expenditures in EU Member States depend on relative contributions of single household expenditures:
data(expendituresEU)
y <- as.numeric(apply(expendituresEU,1,sum))
lmCoDaX(y, expendituresEU, method="classical")
lmCoDaX(y, expendituresEU, method="robust")
```

machineOperators

Compositions of eight-hour shifts of 27 machine operators

Usage

data(machineOperators)

Format

A data frame with 27 observations on the following 4 variables.

Details

- hqproduction high-quality production
- lqproduction low-quality production
- setting machine settings
- repair machine repair


Author(s)

Matthias Templ <matthias.templ@tuwien.ac.at>
manu_abs

References


Examples

```r
data(machineOperators)
str(machineOperators)
summary(machineOperators)
rowSums(machineOperators)
```

manu_abs  Distribution of manufacturing output

Description

The data consists of values of the manufacturing output in 42 countries in 2009. The output, given in national currencies, is structured according to the 3-digit ISIC category and its components. Thorough analysis of the sample is described in Facevicova (2018).

Usage

```r
data(manu_abs)
```

Format

A data frame with 630 observations of 4 variables.

Details

- **country** Country
- **isic** 3-digit ISIC category. The categories are 151 processed meat, fish, fruit, vegetables, fats; 152 Dairy products; 153 Grain mill products, starches, animal feeds; 154 Other food products and 155 Beverages.
- **output** The output components are Labour, Surplus and Input.
- **value** Value of manufacturing output in the national currency

Author(s)

Kamila Facevicova

Source

References


Examples

```r
data(manu_abs)

### Compositional tables approach
### analysis of the relative structure

result <- tabCoordWrapper(manu_abs, obs.ID='country',row.factor = 'output',
col.factor = 'isic', value='value', test = TRUE)

result$Bootstrap

### Classical approach
### generalized linear mixed effect model
## Not run:
library(lme4)
m <- glmer(value~output*as.factor(isic)+(1|country),
data=manu_abs,family=poisson)
summary(m)
## End(Not run)
```

---

**mcad**

*metabolomics mcad data set*

Description

The aim of the experiment was to ascertain novel biomarkers of MCAD (Medium chain acyl-CoA dehydrogenase) deficiency. The data consists of 25 patients and 25 controls and the analysis was done by LC-MS. Rows represent patients and controls and columns represent chemical entities with their quantity.

Usage

```r
data(mcad)
```

Format

A data frame with 50 observations and 279 variables

Details

- group patient group
- ... the remaining variables columns are represented by m/z which are chemical characterizations of individual chemical components on exact mass measurements.
References


Examples

```r
data(mcad)
str(mcad)
```

**missPatterns**

_missing or zero pattern structure._

**Description**

Analysis of the missing or the zero patterns structure of a data set.

**Usage**

```r
missPatterns(x)
zeroPatterns(x)
```

**Arguments**

- `x`: a data frame or matrix.

**Details**

Here, one pattern defines those observations that have the same structure regarding their missingness or zeros. For all patterns a summary is calculated.

**Value**

- `groups`: List of the different patterns and the observation numbers for each pattern.
- `cn`: the names of the patterns coded as vectors of 0-1’s.
- `tabcomb`: the pattern structure - all combinations of zeros or missings in the variables.
- `tabcombPlus`: the pattern structure - all combinations of zeros or missings in the variables including the size of those combinations/patterns, i.e. the number of observations that belongs to each pattern.
- `rsum`: the number of zeros or missing values in each row of the data set.
- `rindex`: the index of zeros or missing values in each row of the data set.
Author(s)
Matthias Templ. The code is based on a previous version from Andreas Alfons and Matthias Templ from package VIM

See Also
aggr

Examples

data(expenditures)
## set NA's artificial:
expenditures[expenditures < 300] <- NA
## detect the NA structure:
missPatterns(expenditures)

Description

- country country name
- country2 country name, short version
- sex gender
- lifeExpectancy life expectancy
- infectious certain infectious and parasitic diseases (A00-B99)
- neoplasms malignant neoplasms (C00-C97)
- endocrine endocrine nutritional and metabolic diseases (E00-E90)
- mental mental and behavioural disorders (F00-F99)
- nervous diseases of the nervous system and the sense organs (G00-H95)
- circulatory diseases of the circulatory system (I00-I99)
- respiratory diseases of the respiratory system (J00-J99)
- digestive diseases of the digestive system (K00-K93)

Usage

data(mortality)

Format
A data frame with 60 observations and 12 variables
Author(s)

Peter Filzmoser, Matthias Templ <matthias,temp@tuwien.ac.at>

References


Examples

data(mortality)
str(mortality)
## totals (mortality)
aggregate(mortality[,5:ncol(mortality)],
      list(mortality$country2), sum)

mortality_tab  mortality table

Description

Mortality data by gender, unknown year

Usage

data(mortality_tab)

Format

A table

Details

- female mortality rates for females by age groups
- male mortality rates for males by age groups

Author(s)

Matthias Templ

Examples

data(mortality_tab)
mortality_tab
nutrients

Description

Nutrients on more than 40 components and 965 generic food products

Usage

data(nutrients)

Format

A data frame with 965 observations on the following 50 variables.

Details

- ID ID, for internal use
- ID_V4 ID V4, for internal use
- ID_SwissFIR ID, for internal use
- name_D Name in German
- name_F Name in French
- name_I Name in Italian
- name_E Name in Spanish
- category_D Category name in German
- category_F Category name in French
- category_I Category name in Italy
- category_E Category name in Spanish
- gravity specific gravity
- 'energy_kJ 'energy in kJ per 100g edible portion
- energy_kcal energy in kcal per 100g edible portion
- protein protein in gram per 100g edible portion
- alcohol alcohol in gram per 100g edible portion
- water water in gram per 100g edible portion
- carbohydrates carbohydrates in gram per 100g edible portion
- starch starch in gram per 100g edible portion
- sugars sugars in gram per 100g edible portion
- 'dietar_fibres 'dietar fibres in gram per 100g edible portion
- fat fat in gram per 100g edible portion
- cholesterol cholesterol in milligram per 100g edible portion
- fattyacids_monounsaturated fatty acids monounsaturated in gram per 100g edible portion
- fattyacids_saturated fatty acids saturated in gram per 100g edible portion
- fatty_acids_polyunsaturated fatty acids polyunsaturated in gram per 100g edible portion
- vitaminA vitamin A in retinol equivalent per 100g edible portion
- all-trans_retinol_equivalents 'all trans-retinol equivalents in gram per 100g edible portion
- beta-carotene-activity 'beta-carotene activity in beta-carotene equivalent per 100g edible portion
- beta-carotene 'beta-carotene in micogram per 100g edible portion
- vitaminB1 vitamin B1 in milligram per 100g edible portion
- vitaminB2 vitamin B2 in milligram per 100g edible portion
- vitaminB6 vitamin B6 in milligram per 100g edible portion
- vitaminB12 vitamin B12 in micogram per 100g edible portion
- niacin niacin in milligram per 100g edible portion
- folate folate in micogram per 100g edible portion
- pantothenic_acid pantothenic acid in milligram per 100g edible portion
- vitaminC vitamin C in milligram per 100g edible portion
- vitaminD vitamin D in micogram per 100g edible portion
- vitaminE vitamin E in alpha-tocopherol equivalent per 100g edible portion
- Na Sodium in milligram per 100g edible portion
- K Potassium in milligram per 100g edible portion
- Ca Calcium
- Mg Magnesium
- P Phosphorus
- Fe Iron
- I Iodide in milligram per 100g edible portion
- Zn Zink
- unit a factor with levels per 100g edible portion per 100ml food volume

Author(s)
Translated from the Swiss nutrition database by Matthias Templ <matthias.templ@tuwien.ac.at>

Source
From the Swiss nutrition data base 2015 (second edition)

Examples

data(nutrients)
str(nutrients)
head(nutrients[, 41:49])
**nutrients_branded**

**nutrient contents (branded)**

**Description**

Nutrients on more than 10 components and 9618 branded food products

**Usage**

data(nutrients_branded)

**Format**

A data frame with 9618 observations on the following 18 variables.

**Details**

- name_D name (in German)
- category_D factor specifying the category names
- category_F factor specifying the category names
- category_I factor specifying the category names
- category_E factor specifying the category names
- gravity specific gravity
- energy_kJ energy in kJ
- `energy_kcal` energy in kcal
- protein protein in gram
- alcohol alcohol in gram
- water water in gram
- carbohydrates_available available carbohydrates in gram
- sugars sugars in gram
- dietary_fibres dietary fibres in gram
- fat_total total fat in gram
- fatty_acids_saturated saturated acids fat in gram
- Na Sodium in gram
- unit a factor with levels per 100g edible portion per 100ml food volume

**Author(s)**

Translated from the Swiss nutrition data base by Matthias Templ <matthias.templ@tuwien.ac.at>

**Source**

From the Swiss nutrition data base 2015 (second edition)
Examples

```r
data(nutrients_branded)
str(nutrients_branded)
```

**orthbasis**  
*Orthonormal basis*

**Description**

Orthonormal basis from cenLR transformed data to pivotCoord transformed data.

**Usage**

```r
orthbasis(D)
```

**Arguments**

- `D` number of parts (variables)

**Details**

For the chosen balances for “pivotCoord”, this is the orthonormal basis that transfers the data from centered logratio to isometric logratio.

**Value**

the orthonormal basis.

**Author(s)**

Karel Hron, Matthias Templ. Some code lines of this function are a copy from function gsi.buildilr from

**See Also**

`pivotCoord`, `cenLR`

**Examples**

```r
data(expenditures)
V <- orthbasis(ncol(expenditures))
xcen <- cenLR(expenditures)$x.clr
xi <- as.matrix(xcen) %*% V$V
xi
xi2 <- pivotCoord(expenditures)
xi2
```
Description
Outlier detection for compositional data using standard and robust statistical methods.

Usage
outCoDa(x, quantile = 0.975, method = "robust", alpha = 0.5, coda = TRUE)

## S3 method for class 'outCoDa'
print(x, ...)

## S3 method for class 'outCoDa'
plot(x, y, ..., which = 1)

Arguments
x compositional data
quantile quantile, corresponding to a significance level, is used as a cut-off value for outlier identification: observations with larger (squared) robust Mahalanobis distance are considered as potential outliers.
method either "robust" (default) or "standard"
alpha the size of the subsets for the robust covariance estimation according the MCD-estimator for which the determinant is minimized, see covMcd.
coda if TRUE, data transformed to coordinate representation before outlier detection.
... additional parameters for print and plot method passed through
y unused second plot argument for the plot method
which 1 ... MD against index 2 ... distance-distance plot

Details
The outlier detection procedure is based on (robust) Mahalanobis distances in isometric logratio coordinates. Observations with squared Mahalanobis distance greater equal a certain quantile of the chi-squared distribution are marked as outliers.

If method “robust” is chosen, the outlier detection is based on the homogeneous majority of the compositional data set. If method “standard” is used, standard measures of location and scatter are applied during the outlier detection procedure.

plot method: the Mahalanobis distance are plotted against the index. The dashed line indicates the (1 - alpha) quantile of the chi-squared distribution. Observations with Mahalanobis distance greater than this quantile could be considered as compositional outliers.
Value

- `mahalDist` resulting Mahalanobis distance
- `limit` quantile of the Chi-squared distribution
- `outlierIndex` logical vector indicating outliers and non-outliers
- `method` method used

Note

It is highly recommended to use the robust version of the procedure.

Author(s)

Matthias Templ, Karel Hron

References


See Also

- `pivotCoord`

Examples

```r
data(expenditures)
oD <- outCoDa(expenditures)
oD
## providing a function:
oD <- outCoDa(expenditures, coda = log)
```

Description

Payments splitted by different NACE categories and kind of employment in Austria 2004

Usage

```r
data(payments)
```
payments

Format

A data frame with 535 rows and 11 variables

Details

- nace NACE classification, 2 digits
- oenace_2008 Corresponding Austrian NACE classification (in German)
- year year
- month month
- localunit local unit ID
- spay special payments (total)
- spay_wc special payments for white collar workers
- spay_bc special payments for blue collar workers
- spay_traintrade special payments for trainees in trade business
- spay_home special payments for home workers
- spay_traincomm special payments for trainees in commercial business

Author(s)

Matthias Templ <matthias.templ@tuwien.ac.at>

Source

statCube data base at the website of Statistics Austria. The product and all material contained therein are protected by copyright with all rights reserved by the Bundesanstalt Statistik Oesterre-ich (STATISTICS AUSTRIA). It is permitted to reproduce, distribute, make publicly available and process the content for non-commercial purposes. Prior to any use for commercial purposes a written consent of STATISTICS AUSTRIA must be obtained. Any use of the contained material must be correctly reproduced and clearly cite the source STATISTICS AUSTRIA. If tables published by STATISTICS AUSTRIA are partially used, displayed or otherwise changed, a note must be added at an adequate position to show data was extracted or adapted.

Examples

data(payments)
str(payments)
snmary(payments)
Robust principal component analysis for compositional data

Description

This function applies robust principal component analysis for compositional data.

Usage

pcaCoDa(
  x,  
  method = "robust",  
  mult_comp = NULL,  
  external = NULL,  
  solve = "eigen"
)

## S3 method for class 'pcaCoDa'
print(x, ...)

## S3 method for class 'pcaCoDa'
summary(object, ...)

Arguments

  x       compositional data
  method  must be either "robust" (default) or "classical"
  mult_comp a list of numeric vectors holding the indices of linked compositions
  external external non-compositional variables
  solve    eigen (as princomp does, i.e. eigenvalues of the covariance matrix) or svd
            (as prcomp does with single value decomposition instead of eigen). Only for
            method classical.
  ...      additional parameters for print method passed through
  object   object of class pcaCoDa

Details

The compositional data set is expressed in isometric logratio coordinates. Afterwards, robust principal component analysis is performed. Resulting loadings and scores are back-transformed to the clr space where the compositional biplot can be shown.

mult_comp is used when there are more than one group of compositional parts in the data. To give an illustrative example, lets assume that one variable group measures angles of the inner ear-bones of animals which sum up to 100 and another one having percentages of a whole on the thickness of the inner ear-bones included. Then two groups of variables exists which are both compositional parts. The isometric logratio coordinates are then internally applied to each group independently whenever the mult_comp is set correctly.
Value

scores scores in clr space
loadings loadings in clr space
eigenvalues eigenvalues of the clr covariance matrix
method method
princompOutputClr

output of princomp needed in plot.pcaCoDa

Author(s)

Karel Hron, Peter Filzmoser, Matthias Templ and a contribution for dimnames in external variables by Amelia Landre.

References


See Also

print.pcaCoDa, summary.pcaCoDa, biplot.pcaCoDa, plot.pcaCoDa

Examples

data(arcticLake)

## robust estimation (default):
res.rob <- pcaCoDa(arcticLake)
res.rob
summary(res.rob)
plot(res.rob)

## classical estimation:
res.cla <- pcaCoDa(arcticLake, method="classical", solve = "eigen")
biplot(res.cla)

## just for illustration how to set the mult_comp argument:
data(expenditures)
p1 <- pcaCoDa(expenditures, mult_comp=list(c(1,2,3),c(4,5)))
p1

## example with external variables:
data(election)
# transform external variables
election$unemployment <- log((election$unemployment/100)/(1-election$unemployment/100))
election$income <- scale(election$income)
perturbation

Perturbation and powering for two compositions.

Usage

perturbation(x, y)

powering(x, a)

Arguments

x (compositional) vector containing positive values
y (compositional) vector containing positive values or NULL for powering
a constant, numeric vector of length 1

Value

Result of perturbation or powering

Author(s)

Matthias Templ

References


Examples

data(expenditures)
x <- expenditures[1 ,]y <- expenditures[2, ]perturbation(x, y)powering(x, 2)
pfa

Factor analysis for compositional data

Description
Computes the principal factor analysis of the input data which are transformed and centered first.

Usage
pfa(
  x,
  factors,
  robust = TRUE,
  data = NULL,
  covmat = NULL,
  n.obs = NA,
  subset,
  na.action,
  start = NULL,
  scores = c("none", "regression", "Bartlett"),
  rotation = "varimax",
  maxiter = 5,
  control = NULL,
  ...
)

Arguments
x (robustly) scaled input data
factors number of factors
robust default value is TRUE
data default value is NULL
covmat (robustly) computed covariance or correlation matrix
n.obs number of observations
subset if a subset is used
na.action what to do with NA values
start starting values
scores which method should be used to calculate the scores
rotation if a rotation should be made
maxiter maximum number of iterations
control default value is NULL
... arguments for creating a list
Details

The main difference to usual implementations is that uniquenesses are no longer of diagonal form. This kind of factor analysis is designed for centered log-ratio transformed compositional data. However, if the covariance is not specified, the covariance is estimated from isometric log-ratio transformed data internally, but the data used for factor analysis are backtransformed to the clr space (see Filzmoser et al., 2009).

Value

loadings A matrix of loadings, one column for each factor. The factors are ordered in decreasing order of sums of squares of loadings.

uniqueness uniqueness

correlation correlation matrix

criteria The results of the optimization: the value of the negative log-likelihood and information of the iterations used.

factors the factors
dof degrees of freedom

method “principal”
n.obs number of observations if available, or NA

call The matched call.

STATISTIC, PVAL

The significance-test statistic and p-value, if they can be computed

Author(s)

Peter Filzmoser, Karel Hron, Matthias Templ

References


Examples

data(expenditures)
x <- expenditures
res.rob <- pfa(x, factors=1)
res.cla <- pfa(x, factors=1, robust=FALSE)

## the following produce always the same result:
res1 <- pfa(x, factors=1, covmat="covMcd")
res2 <- pfa(x, factors=1, covmat=robustbase::covMcd(pivotCoord(x)))$cov
res3 <- pfa(x, factors=1, covmat=robustbase::covMcd(pivotCoord(x)))
Description

PhD students in Europe based on the standard classification system splitted by different kind of studies (given as percentages).

Format

A data set on 32 compositions and 11 variables.

Details

Due to unknown reasons the rowSums of the percentages is not always 100.

- country country of origin (German)
- countryEN country of origin (English)
- country2 country of origin, 2-digits
- total total phd students (in 1.000)
- male male phd students (in 1.000)
- female total phd students (in 1.000)
- technical phd students in natural and technical sciences
- socio-economic-low phd students in social sciences, economic sciences and law sciences
- human phd students in human sciences including teaching
- health phd students in health and life sciences
- agriculture phd students in agriculture

Source

Eurostat

References


Examples

data(phd)
str(phd)
### Description
PhD students in Europe by different kind of studies.

### Format
A data set on 29 compositions and 5 variables.

### Details
- technical phd students in natural and technical sciences
- socio-economic-low phd students in social sciences, economic sciences and law sciences
- human phd students in human sciences including teaching
- health phd students in health and life sciences
- agriculture phd students in agriculture

### Source
Eurostat

### References

### Examples
```r
data("phd_totals")
str(phd_totals)
```
pivotCoord

Pivot coordinates and their inverse

Description

Pivot coordinates as a special case of isometric logratio coordinates and their inverse mapping.

Usage

```r
pivotCoord(
  x,
  pivotvar = 1,
  fast = FALSE,
  method = "pivot",
  base = exp(1),
  norm = "orthonormal"
)
```

```r
isomLR(x, fast = FALSE, base = exp(1), norm = "sqrt((D-i)/(D-i+1))")
```

```r
isomLRinv(x)
```

```r
pivotCoordInv(x, norm = "orthonormal")
```

```r
isomLRp(x, fast = FALSE, base = exp(1), norm = "sqrt((D-i)/(D-i+1))")
```

```r
isomLRinvp(x)
```

Arguments

- **x**: object of class data.frame or matrix. Positive values only.
- **pivotvar**: pivotal variable. If any other number than 1, the data are resorted in that sense that the pivotvar is shifted to the first part.
- **fast**: if TRUE, it is approx. 10 times faster but numerical problems in case of high-dimensional data may occur. Only available for method “pivot”.
- **method**: pivot takes the method described in the description. Method "symm" uses symmetric pivot coordinates (parameters pivotvar and norm have then no effect)
- **base**: a positive or complex number: the base with respect to which logarithms are computed. Defaults to exp(1).
- **norm**: if FALSE then the normalizing constant is not used, if TRUE sqrt(((D-i)/(D-i+1))) is used (default). The user can also specify a self-defined constant.
Details

Pivot coordinates map D-part compositional data from the simplex into a (D-1)-dimensional real space isometrically. From our choice of pivot coordinates, all the relative information about one of parts (or about two parts) is aggregated in the first coordinate (or in the first two coordinates in case of symmetric pivot coordinates, respectively).

Value

The data represented in pivot coordinates

Author(s)

Matthias Templ, Karel Hron, Peter Filzmoser

References


Examples

```r
require(MASS)
Sigma <- matrix(c(5.05,4.95,4.95,5.05), ncol=2, byrow=TRUE)
z <- pivotCoordInv(mvrnorm(100, mu=c(0,2), Sigma=Sigma))

data(expenditures)
## first variable as pivot variable
pivotCoord(expenditures)
## third variable as pivot variable
pivotCoord(expenditures, 3)

x <- exp(mvrnorm(2000, mu=rep(1,10), diag(10)))
system.time(pivotCoord(x))
system.time(pivotCoord(x, fast=TRUE))

## without normalizing constant
pivotCoord(expenditures, norm = "orthogonal") # or:
pivotCoord(expenditures, norm = "1")
## other normalization
pivotCoord(expenditures, norm = "-sqrt((D-i)/(D-i+1))")

# symmetric balances (results in 2-dim symmetric pivot coordinates)
pivotCoord(expenditures, method = "symm")
```
plot.imp

Plot method for objects of class imp

Description

This function provides several diagnostic plots for the imputed data set in order to see how the imputed values are distributed in comparison with the original data values.

Usage

## S3 method for class 'imp'
plot(
x, ...
which = 1,
ord = 1:ncol(x),
colcomb = "missnonmiss",
plotvars = NULL,
col = c("skyblue", "red"),
alpha = NULL,
lty = par("lty"),
xaxt = "s",
xaxlabels = NULL,
las = 3,
interactive = TRUE,
pch = c(1, 3),
ask = prod(par("mfcol")) < length(which) && dev.interactive(),
center = FALSE,
scale = FALSE,
id = FALSE,
seg.l = 0.02,
seg1 = TRUE
)

Arguments

x  object of class ‘imp’
...
which   if a subset of the plots is required, specify a subset of the numbers 1:3.
ord   determines the ordering of the variables
colcomb   if colcomb="missnonmiss", observations with missings in any variable are highlighted. Otherwise, observations with missings in any of the variables specified by colcomb are highlighted in the parallel coordinate plot.
plotvars   Parameter for the parallel coordinate plot. A vector giving the variables to be plotted. If NULL (the default), all variables are plotted.
col  a vector of length two giving the colors to be used in the plot. The second color will be used for highlighting.

alpha a numeric value between 0 and 1 giving the level of transparency of the colors, or NULL. This can be used to prevent overplotting.

lty  a vector of length two giving the line types. The second line type will be used for the highlighted observations. If a single value is supplied, it will be used for both non-highlighted and highlighted observations.

xaxt the x-axis type (see par).

xaxlabels a character vector containing the labels for the x-axis. If NULL, the column names of x will be used.

las  the style of axis labels (see par).

interactive a logical indicating whether the variables to be used for highlighting can be selected interactively (see ‘Details’).

pch  a vector of length two giving the symbol of the plotting points. The symbol will be used for the highlighted observations. If a single value is supplied, it will be used for both non-highlighted and highlighted observations.

ask logical; if TRUE, the user is asked before each plot, see par(ask=).

center logical, indicates if the data should be centered prior plotting the ternary plot.

scale logical, indicates if the data should be centered prior plotting the ternary plot.

id reads the position of the graphics pointer when the (first) mouse button is pressed and returns the corresponding index of the observation. (only used by the ternary plot)

seg.l length of the plotting symbol (spikes) for the ternary plot.

seg1 if TRUE, the spikes of the plotting symbol are justified.

Details

The first plot (which == 1) is a multiple scatterplot where for the imputed values another plot symbol and color is used in order to highlight them. Currently, the ggpairs functions from the GGally package is used.

Plot 2 is a parallel coordinate plot in which imputed values in certain variables are highlighted. In parallel coordinate plots, the variables are represented by parallel axes. Each observation of the scaled data is shown as a line. If interactive is TRUE, the variables to be used for highlighting can be selected interactively. Observations which includes imputed values in any of the selected variables will be highlighted. A variable can be added to the selection by clicking on a coordinate axis. If a variable is already selected, clicking on its coordinate axis will remove it from the selection. Clicking anywhere outside the plot region quits the interactive session.

Plot 3 shows a ternary diagram in which imputed values are highlighted, i.e. those spikes of the chosen plotting symbol are colored in red for which of the values are missing in the unimputed data set.

Value

None (invisible NULL).
plot.pcaCoDa

Description
Provides a screeplot and biplot for (robust) compositional principal components analysis.

Usage
```r
## S3 method for class 'pcaCoDa'
plot(x, y, ..., which = 1, choices = 1:2)
```

Arguments
- **x**: object of class ‘pcaCoDa’
- **y**: ...
- **...**: ...
- **which**: an integer between 1 and 3. Produces a screeplot (1), or a biplot using stats biplot.prcomp function (2), or a biplot using ggfortify’s autoplot function (3).
- **choices**: principal components to plot by number

Author(s)
Matthias Templ

References


See Also
impCoda, impKNNa

Examples
```r
data(expenditures)
expenditures[1,3]
expenditures[1,3] <- NA
xi <- impKNNa(expenditures)
xi
summary(xi)
## Not run: plot(xi, which=1)
plot(xi, which=2)
plot(xi, which=3)
plot(xi, which=3, segl=FALSE)
```

plot.pcaCoDa

Plot method

```r
plot.pcaCoDa
```
Value

The robust compositional screeplot.

Author(s)

M. Templ, K. Hron

References


See Also

*pcaCoDa*, *biplot.pcaCoDa*

Examples

data(coffee)
## Not run:
p1 <- pcaCoDa(coffee[, -1])
plot(p1)
plot(p1, type = "lines")
plot(p1, which = 2)
plot(p1, which = 3)
## End(Not run)

plot.smoothSpl

plot densities of objects of class smoothSpl

Usage

## S3 method for class 'smoothSpl'
plot(x, y, ..., by = 1, n = 10, index = NULL)
Arguments

- `x`: class smoothSpl object
- `y`: ignored
- `...`: further arguments passed by `by`
- `n`: steps size
- `index`: length of sequence to plot
- `...`: optionally the sequence instead of `by` and `n`

Author(s)

Alessia Di Blasi, Federico Pavone, Gianluca Zeni

Data Frame

```
              precipitation
24-hour precipitation
```

Description

table containing counts for 24-hour precipitation for season at the rain-gouge.

Usage

data(precipitation)

Format

A table with 4 rows and 6 columns

Details

- `spring`: numeric vector on counts for different level of precipitation
- `summer`: numeric vector on counts for different level of precipitation
- `autumn`: numeric vector on counts for different level of precipitation
- `winter`: numeric vector on counts for different level of precipitation

Author(s)

Matthias Templ <matthias.templ@tuwien.ac.at>

References

Examples

```r
data(precipitation)
precipitation
str(precipitation)
```

---

**print.imp**

*Print method for objects of class imp*

---

**Description**

The function returns a few information about how many missing values are imputed and possible other information about the amount of iterations, for example.

**Usage**

```r
## S3 method for class 'imp'
print(x, ...)
```

**Arguments**

- `x` an object of class ‘imp’
- `...` additional arguments passed trough

**Value**

None (invisible NULL).

**Author(s)**

Matthias Templ

**See Also**

`impCoda, impKNNa`

**Examples**

```r
data(expenditures)
expenditures[1,3]
expenditures[1,3] <- NA
## Not run:
xi <- impCoda(expenditures)
xi
summary(xi)
plot(xi, which=1:2)

## End(Not run)
```
**production**

*production splitted by nationality on enterprise level*

**Description**

- nace NACE classification, 2 digits
- oenace_2008 Corresponding Austrian NACE classification (in German)
- year
- month
- enterprise enterprise ID
- total total ...
- home home ...
- EU EU ...
- non-EU non-EU ...

**Usage**

```r
data(production)
```

**Format**

A data frame with 535 rows and 9 variables

**Author(s)**

Matthias Templ <matthias.templ@tuwien.ac.at>

**Source**

statCube data base at the website of Statistics Austria. The product and all material contained therein are protected by copyright with all rights reserved by the Bundesanstalt Statistik Österreich (STATISTICS AUSTRIA). It is permitted to reproduce, distribute, make publicly available and process the content for non-commercial purposes. Prior to any use for commercial purposes a written consent of STATISTICS AUSTRIA must be obtained. Any use of the contained material must be correctly reproduced and clearly cite the source STATISTICS AUSTRIA. If tables published by STATISTICS AUSTRIA are partially used, displayed or otherwise changed, a note must be added at an adequate position to show data was extracted or adapted.

**Examples**

```r
data(production)
str(production)
summary(production)
```
pTab  

Propability table

Description

Calculates the propability table using different methods

Usage

pTab(x, method = "dirichlet", alpha = 1/length(as.numeric(x)))

Arguments

x an object of class table

method default is ‘dirichlet’. Other available methods: ‘classical’ that is function prop.table() from package base or method “half” that add 1/2 to each cell to avoid zero problems.

alpha constant used for method ‘dirichlet’

Value

The probablity table

Author(s)

Matthias Templ

References


Examples

data(precipitation)
pTab(precipitation)
pTab(precipitation, method = "dirichlet")
rcodes codes for UNIDO tables

Description

- ISOCNISOCON codes
- OPERATOROperator
- ADESCCountry
- CCODECountry code
- CDESCCountry destination
- ACODECountry destination code

Usage

data(rcodes)

Format

A data.frame with 2717 rows and 6 columns.

Examples

data(rcodes)
str(rcodes)

rdcm relative difference between covariance matrices

Description

The sample covariance matrices are computed from compositions expressed in the same isometric logratio coordinates.

Usage

rdcm(x, y)

Arguments

- x matrix or data frame
- y matrix or data frame of the same size as x.
Details

The difference in covariance structure is based on the Euclidean distance between both covariance estimations.

Value

the error measures value

Author(s)

Matthias Templ

References


See Also

rdcm

Examples

data(expenditures)
x <- expenditures
x[1,3] <- NA
xi <- impKNNa$xImp
rdcm(expenditures, xi)

---

rSDev

| Relative simplicial deviance |

Description

Relative simplicial deviance

Usage

rSDev(x, y)

Arguments

- **x**: a probability table
- **y**: an interaction table
**rSDev.test**

**Value**
The relative simplicial deviance

**Author(s)**
Matthias Templ

**References**
tables using simplicial geometry. *Communications in Statistics - Theory and Methods*, 44 (18),
3978–3996.

**Examples**
```r
data(precipitation)
tabprob <- prop.table(precipitation)
tabind <- indTab(precipitation)
tabint <- intTab(tabprob, tabind)
rSDev(tabprob, tabint$intTab)
```

**rSDev.test**

**Relative simplicial deviance tests**

**Description**
Monte Carlo based contingency table tests considering the compositional approach to contingency
tables.

**Usage**
rSDev.test(x, R = 999, method = "multinom")

**Arguments**
- **x**: matrix, data.frame or table
- **R**: an integer specifying the number of replicates used in the Monte Carlo test.
- **method**: either “rmultinom” (default) or “permutation”.

**Details**
Method “rmultinom” generate multinomially distributed samples from the independent probability
table, which is estimated from x using geometric mean marginals. The relative simplicial deviance
of the original data are then compared to the generated ones.
Method “permutation” permutes the entries of x and compares the relative simplicial deviance esti-
mated from the original data to the ones of the permuted data (the independence table is unchanged
and originates on x).
Method “rmultinom” should be preferred, while method “permutation” can be used for compar-
isons.
Value
A list with class “htest” containing the following components:

- statistic: the value of the relative simplicial deviance (test statistic).
- method: a character string indicating what type of rSDev.test was performed.
- p.value: the p-value for the test.

Author(s)
Matthias Templ, Karel Hron

References

See Also
rSDev

Examples
data(precipitation)
rSDev.test(precipitation)

---

saffron compositions

Description
Stable isotope ratio and trace metal concentration data for saffron samples.

Format
A data frame with 53 observations on the following 36 variables.

- Sample: adulterated honey, Honey or Syrup
- Country: group information
- Batch: detailed group information
- Region: less detailed group information
- d2H: region
- d13C: chemical element
- d15N: chemical element
- Li: chemical element
• B chemical element
• Na chemical element
• Mg chemical element
• Al chemical element
• K chemical element
• Ca chemical element
• V chemical element
• Mn chemical element
• Fe chemical element
• Co chemical element
• Ni chemical element
• Cu chemical element
• Zn chemical element
• Ga chemical element
• As chemical element
• Rb chemical element
• Sr chemical element
• Y chemical element
• Mo chemical element
• Cd chemical element
• Cs chemical element
• Ba chemical element
• Ce chemical element
• Pr chemical element
• Nd chemical element
• Sm chemical element
• Gd chemical element
• Pb chemical element

Note
In the original paper, the authors applied LDA for classifying the observations.

Source
Mendeley Data, contributed by Russell Frew and translated to R by Matthias Templ

References
Frew, Russell (2019), Data for: CHEMICAL PROFILING OF SAFFRON FOR AUTHENTICATION OF ORIGIN, Mendeley Data, V1, doi: 10.17632/5544tn9v6c.1
Examples

```
data(saffron)
```

<table>
<thead>
<tr>
<th>SDev</th>
<th>Simplicial deviance</th>
</tr>
</thead>
</table>

Description

Simplicial deviance

Usage

```
SDev(x)
```

Arguments

```
x             a probability table
```

Value

The simplicial deviance

Author(s)

Matthias Templ

References


Examples

```
data(precipitation)
tablprob <- prop.table(precipitation)
SDev(tablprob)
```
AFM compositions of 23 aphyric Skye lavas. This data set can be found on page 360 of the Aitchison book (see reference).

Usage

data(skyeLavas)

Format

A data frame with 23 observations on the following 3 variables.

Details

- sodium-potassium a numeric vector of percentages of Na2O+K2O
- iron a numeric vector of percentages of Fe2O3
- magnesium a numeric vector of percentages of MgO

Author(s)

Matthias Templ <matthias.templ@tuwien.ac.at>

References


Examples

data(skyeLavas)
str(skyeLavas)
summary(skyeLavas)
rowSums(skyeLavas)
smoothSplines

Description

Given raw (discretized) distributional observations, smoothSplines computes the density function that 'best' fits data, in a trade-off between smooth and least squares approximation, using B-spline basis functions.

Usage

smoothSplines(
  k, l, alpha, data, xcp, knots,
  weights = matrix(1, dim(data)[1], dim(data)[2]),
  num_points = 100,
  prior = "default",
  cores = 1,
  fast = 0
)

Arguments

k       smoothing splines degree
l       order of derivative in the penalization term
alpha   weight for penalization
data     an object of class "matrix" containing data to be smoothed, row by row
xcp     vector of control points
knots   either vector of knots for the splines or a integer for the number of equispaced knots
weights matrix of weights. If not given, all data points will be weighted the same.
num_points number of points of the grid where to evaluate the density estimated
prior   prior used for zero-replacements. This must be one of "perks", "jeffreys", "bayes_laplace", "sq" or "default"
cores   number of cores for parallel execution, if the option was enabled before installing the package
fast    1 if maximal performance is required (print statements suppressed), 0 otherwise
Details

The original discretized densities are not directly smoothed, but instead the centred logratio transformation is first applied, to deal with the unit integral constraint related to density functions. Then the constrained variational problem is set. This minimization problem for the optimal density is a compromise between staying close to the given data, at the corresponding $x_{cp}$, and obtaining a smooth function. The non-smoothness measure takes into account the 1th derivative, while the fidelity term is weighted by $\alpha$.

The solution is a natural spline. The vector of its coefficients is obtained by the minimum norm solution of a linear system. The resulting splines can be either back-transformed to the original Bayes space of density functions (in order to provide their smoothed counterparts for visualization and interpretation purposes), or retained for further statistical analysis in the clr space.

Value

An object of class `smoothSpl`, containing among the other the following variables:

- `bspline`: each row is the vector of B-spline coefficients
- `Y`: the values of the smoothed curve, for the grid given
- `Y_clr`: the values of the smoothed curve, in the clr setting, for the grid given

Author(s)

Alessia Di Blasi, Federico Pavone, Gianluca Zeni, Matthias Templ

References


Examples

```r
SepalLengthCm <- iris$Sepal.Length
Species <- iris$Species

iris1 <- SepalLengthCm[iris$Species==levels(iris$Species)[1]]
h1 <- hist(iris1, nclass = 12, plot = FALSE)

midx1 <- h1$mids
midy1 <- matrix(h1$density, nrow=1, ncol = length(h1$density), byrow=TRUE)
knots <- 7

## Not run:
sol1 <- smoothSplines(k=3,l=2,alpha=1000,midy1,midx1,knots)
plot(sol1)

h1 <- hist(iris1, freq = FALSE, nclass = 12, xlab = "Sepal Length [cm]", main = "Iris setosa")
# black line: kernel method; red line: smoothSplines result
lines(density(iris1), col = "black", lwd = 1.5)
xx1 <- seq(sol1$Xcp[1],tail(sol1$Xcp,n=1),length.out = sol1$NumPoints)
lines(xx1,sol1$Y[1,], col = "red", lwd = 2)

## End(Not run)
```
smoothSplinesVal

Estimate density from histogram - for different alpha

Description

As smoothSplines, smoothSplinesVal computes the density function that 'best' fits discretized distributional data, using B-spline basis functions, for different alpha. Comparing and choosing an appropriate alpha is the ultimate goal.

Usage

smoothSplinesVal(
  k,
  l,
  alpha,
  data,
  xcp,
  knots,
  weights = matrix(1, dim(data)[1], dim(data)[2]),
  prior = "default",
  cores = 1
)

Arguments

k    smoothing splines degree
l    order of derivative in the penalization term
alpha    vector of weights for penalization
data    an object of class "matrix" containing data to be smoothed, row by row
xcp    vector of control points
knots    either vector of knots for the splines or a integer for the number of equispaced knots
weights    matrix of weights. If not gives, all data points will be weighted the same.
prior    prior used for zero-replacements. This must be one of "perks", "jeffreys", "bayes_laplace", "sq" or "default"
cores    number of cores for parallel execution

Details

See smoothSplines for the description of the algorithm.
Value

A list of three objects:

- \( \alpha \) the values of \( \alpha \)
- \( J \) the values of the functional evaluated in the minimizing
- \( CV\text{-error} \) the values of the leave-one-out \( CV\text{-error} \)

Author(s)

Alessia Di Blasi, Federico Pavone, Gianluca Zeni, Matthias Templ

References


Examples

```r
SepalLengthCm <- iris$Sepal.Length
Species <- iris$Species

iris1 <- SepalLengthCm[iris$Species==levels(iris$Species)[1]]
h1 <- hist(iris1, nclass = 12, plot = FALSE)

## Not run:
midx1 <- h1$mids
midy1 <- matrix(h1$density, nrow=1, ncol = length(h1$density), byrow=TRUE)
knots <- 7
sol1 <- smoothSplinesVal(k=3,l=2,alpha=10^seq(-4,4,by=1),midy1,midx1,knots,cores=1)

## End(Not run)
```

Description

Social expenditures according to source (public or private) and three important branches (health, old age, incapacity related) in selected OECD countries in 2010. Expenditures are always provided in the respective currency.

Usage

data(socExp)

Format

A data frame with 20 observations on the following 8 variables (country + currency + row-wise sorted cells of 2x3 compositional table).
Details

- country Country of origin
- currency Currency unit (in Million)
- health-public Health from the public
- old-public Old age expenditures from the public
- incap-public Incapacity related expenditures from the public
- health-private Health from private sources
- old-private Old age expenditures from private sources
- incap-private Incapacity related expenditures from private sources

Author(s)

conversion to R by Karel Hron Karel Hron and modifications by Matthias Templ <matthias.templ@tuwien.ac.at>

References

OECD, https://www.oecd.org

Examples

data(socExp)
str(socExp)
rowSums(socExp[, 3:ncol(socExp)])

---

stats  Classical estimates for tables

Description

Some standard/classical (non-compositional) statistics

Usage

stats(
   x,
   margins = NULL,
   statistics = c("phi", "cramer", "chisq", "yates"),
   maggr = mean
)
Arguments

x a data.frame, matrix or table
margins margins
statistics statistics of interest
maggr a function for calculating the mean margins of a table, default is the arithmetic mean

Details

statistics ‘phi’ is the values of the table divided by the product of margins. ‘cramer’ normalize these values according to the dimension of the table. ‘chisq’ are the expected values according to Pearson while ‘yates’ according to Yates.

For the maggr function argument, arithmetic means (mean) should be chosen to obtain the classical results. Any other user-provided functions should be take with care since the classical estimations relies on the arithmetic mean.

Value

List containing all statistics

Author(s)

Matthias Templ

References


Examples

data(precipitation)
tab1 <- indTab(precipitation)
stats(precipitation)
stats(precipitation, statistics = "cramer")
stats(precipitation, statistics = "chisq")
stats(precipitation, statistics = "yates")

## take with care
## (the provided statistics are not designed for that case):
stats(precipitation, statistics = "chisq", maggr = gmean)
Summary method for objects of class imp

Description
A short comparison of the original data and the imputed data is given.

Usage
## S3 method for class 'imp'
summary(object, ...)

Arguments
object an object of class ‘imp’
... additional arguments passed through

Details
Note that this function will be enhanced with more sophisticated methods in future versions of the package. It is very rudimental in its present form.

Value
None (invisible NULL).

Author(s)
Matthias Templ

See Also
impCoda, impKNNa

Examples
data(expenditures)
expenditures[1,3]
expenditures[1,3] <- NA
xi <- impKNNa(expenditures)
xi
summary(xi)
# plot(xi, which=1:2)
**Description**

`tabCoord` computes a system of orthonormal coordinates of a compositional table. Computation of either pivot coordinates or a coordinate system based on the given SBP is possible.

`tabCoordWrapper`: For each compositional table in the sample `tabCoordWrapper` computes a system of orthonormal coordinates and provide a simple descriptive analysis. Computation of either pivot coordinates or a coordinate system based on the given SBP is possible.

**Usage**

```r
tabCoord(
  x = NULL,
  row.factor = NULL,
  col.factor = NULL,
  value = NULL,
  SBPr = NULL,
  SBPc = NULL,
  pivot = FALSE,
  print.res = FALSE
)

tabCoordWrapper(
  X,
  obs.ID = NULL,
  row.factor = NULL,
  col.factor = NULL,
  value = NULL,
  SBPr = NULL,
  SBPc = NULL,
  pivot = FALSE,
  test = FALSE,
  n.boot = 1000
)
```

**Arguments**

- **x** a data frame containing variables representing row and column factors of the respective compositional table and variable with the values of the composition.
- **row.factor** name of the variable representing the row factor. Needs to be stated with the quotation marks.
- **col.factor** name of the variable representing the column factor. Needs to be stated with the quotation marks.
value  name of the variable representing the values of the composition. Needs to be stated with the quotation marks.

SBPr an $I - 1 \times I$ array defining the sequential binary partition of the values of the row factor, where $I$ is the number of the row factor levels. The values assigned in the given step to the + group are marked by 1, values from the - group by -1 and the rest by 0. If it is not provided, the pivot version of coordinates is constructed automatically.

SBPc an $J - 1 \times J$ array defining the sequential binary partition of the values of the column factor, where $J$ is the number of the column factor levels. The values assigned in the given step to the + group are marked by 1, values from the - group by -1 and the rest by 0. If it is not provided, the pivot version of coordinates is constructed automatically.

pivot logical, default is FALSE. If TRUE, or one of the SBPs is not defined, its pivot version is used.

print.res logical, default is FALSE. If TRUE, the output is displayed in the Console.

X a data frame containing variables representing row and column factors of the respective compositional tables, variable with the values of the composition and variable distinguishing the observations.

obs.ID name of the variable distinguishing the observations. Needs to be stated with the quotation marks.

test logical, default is FALSE. If TRUE, the bootstrap analysis of coordinates is provided.

n.boot number of bootstrap samples.

Details

tabCoord
This transformation moves the IJ-part compositional tables from the simplex into a (IJ-1)-dimensional real space isometrically with respect to its two-factorial nature. The coordinate system is formed by two types of coordinates - balances and log odds-ratios.

tabCoordWrapper: Each of n IJ-part compositional tables from the sample is with respect to its two-factorial nature isometrically transformed from the simplex into a (IJ-1)-dimensional real space. Sample mean values and standard deviations are computed and using bootstrap an estimate of 95% confidence interval is given.

Value

Coordinates an array of orthonormal coordinates.

Grap.rep graphical representation of the coordinates. Parts denoted by + form the groups in the numerator of the respective computational formula, parts - form the denominator and parts . are not involved in the given coordinate.

Ind.coord an array of row and column balances. Coordinate representation of the independent part of the table.

Int.coord an array of OR coordinates. Coordinate representation of the interactive part of the table.
Contrast.matrix
contrast matrix.

Log.ratios
an array of pure log-ratios between groups of parts without the normalizing constant.

Coda.table
table form of the given composition.

Bootstrap
array of sample means, standard deviations and bootstrap confidence intervals.

Tables
Table form of the given compositions.

Author(s)
Kamila Facevicova

References

See Also
cubeCoord cubeCoordWrapper

Examples

# Coordinate representation of a CoDa Table
# example from Fačevićova (2018):
data(manu_abs)
manu_USA <- manu_abs[which(manu_abs$country=='USA'),]
manu_USA$output <- factor(manu_USA$output, levels=c('LAB', 'SUR', 'INP'))

# pivot coordinates
tabCoord(manu_USA, row.factor = 'output', col.factor = 'isic', value='value')

# SBPs defined in paper
r <- rbind(c(-1,-1,1), c(-1,1,0))
c <- rbind(c(-1,-1,-1,-1,1), c(-1,-1,-1,1,0), c(-1,-1,1,0,0), c(-1,1,0,0,0))
tabCoord(manu_USA, row.factor = 'output', col.factor = 'isic', value='value', SBPr=r, SBPc=c)

# Analysis of a sample of CoDa Tables
# example from Fačevićova (2018):
data(manu_abs)

### Compositional tables approach,
### analysis of the relative structure.
### An example from Facevicova (2018)

manu_abs$output <- factor(manu_abs$output, levels=c('LAB', 'SUR', 'INP'))
# pivot coordinates
```
tabCoordWrapper(manu_abs, obs.ID='country',
row.factor = 'output', col.factor = 'isic', value='value')
```

# SBPs defined in paper
```
r <- rbind(c(-1,-1,1), c(-1,1,0))
c <- rbind(c(-1,-1,-1,-1,1), c(-1,-1,-1,1,0),
c(-1,-1,1,0,0), c(-1,1,0,0,0))
tabCoordWrapper(manu_abs, obs.ID='country', row.factor = 'output',
col.factor = 'isic', value='value', SBPr=r, SBPc=c, test=TRUE)
```

### Classical approach,
### generalized linear mixed effect model.

## Not run:
```
library(lme4)
glmer(value~output*as.factor(isic)+(1|country),data=manu_abs,family=poisson)
```

## End(Not run)

---

**teachingStuff**

### Description
Teaching stuff in selected countries

### Format
A (tidy) data frame with 1216 observations on the following 4 variables.

- **country** Country of origin
- **subject** school type: primary, lower secondary, higher secondary and tertiary
- **year** Year
- **value** Number of stuff

### Details
Teaching staff include professional personnel directly involved in teaching students, including classroom teachers, special education teachers and other teachers who work with students as a whole class, in small groups, or in one-to-one teaching. Teaching staff also include department chairs of whose duties include some teaching, but it does not include non-professional personnel who support teachers in providing instruction to students, such as teachers’ aides and other paraprofessional personnel. Academic staff include personnel whose primary assignment is instruction, research or public service, holding an academic rank with such titles as professor, associate professor, assistant professor, instructor, lecturer, or the equivalent of any of these academic ranks. The category includes personnel with other titles (e.g. dean, director, associate dean, assistant dean, chair or head of department), if their principal activity is instruction or research.
**ternaryDiag**

**Author(s)**
translated from https://data.oecd.org/ and restructured by Matthias Templ

**Source**
OECD: https://data.oecd.org/

**References**

**Examples**

```r
data(teachingStuff)
str(teachingStuff)
```

---

ternaryDiag  Ternary diagram

---

**Description**
This plot shows the relative proportions of three variables (compositional parts) in one diagramm. Before plotting, the data are scaled.

**Usage**

```r
ternaryDiag(
  x,
  name = colnames(x),
  text = NULL,
  grid = TRUE,
  gridCol = grey(0.6),
  mcex = 1.2,
  line = "none",
  robust = TRUE,
  group = NULL,
  tol = 0.975,
  ...
)
```

**Arguments**

- **x** matrix or data.frame with 3 columns
- **name** names of the variables
- **text** default NULL, text for each point can be provided
- **grid** if TRUE a grid is plotted additionally in the ternary diagram
gridCol: color for the grid lines
ncex: label size
line: may be set to “none”, “pca”, “regression”, “regressionconf”, “regressionpred”, “ellipse”, “lda”
robust: if line equals TRUE, it dedicates if a robust estimation is applied or not.
group: if line equals “da”, it determines the grouping variable
tol: if line equals “ellipse”, it determines the parameter for the tolerance ellipse
...: further parameters, see, e.g., `par()`

Details

The relative proportions of each variable are plotted.

Author(s)

Peter Filzmoser <P.Filzmoser@tuwien.ac.at>, Matthias Templ

References


See Also

ternary

Examples

data(arcticLake)
ternaryDiag(arcticLake)

data(coffee)
x <- coffee[,2:4]
grp <- as.integer(coffee[,1])
ternaryDiag(x, col=grp, pch=grp)
ternaryDiag(x, grid=FALSE, col=grp, pch=grp)
legend("topright", legend=unique(coffee[,4]), pch=1:2, col=1:2)

ternaryDiag(x, grid=FALSE, col=grp, pch=grp, line="ellipse", tol=c(0.975,0.9), lty=2)
ternaryDiag(x, grid=FALSE, line="pca")
ternaryDiag(x, grid=FALSE, col=grp, pch=grp, line="pca", lty=2, lwd=2)
ternaryDiagAbline

Adds a line to a ternary diagram.

Description

A low-level plot function which adds a line to a high-level ternary diagram.

Usage

ternaryDiagAbline(x, ...)

Arguments

x Two-dimensional data set in isometric log-ratio transformed space.
...
Additionnal graphical parameters passed through.

Details

This is a small utility function which helps to add a line in a ternary plot from two given points in an isometric transformed space.

Value

no values are returned.

Author(s)

Matthias Templ

See Also

ternaryDiag

Examples

data(coffee)
x <- coffee[,2:4]
ternaryDiag(x, grid=FALSE)
ternaryDiagAbline(data.frame(z1=c(0.01,0.5), z2=c(0.4,0.8)), col="red")
ternaryDiagEllipse  

_Adds tolerance ellipses to a ternary diagram._

**Description**

Low-level plot function which add tolerance ellipses to a high-level plot of a ternary diagram.

**Usage**

ternaryDiagEllipse(x, tolerance = c(0.9, 0.95, 0.975), locscatt = "MCD", ...)  

**Arguments**

- **x**: Three-part composition. Object of class “matrix” or “data.frame”.
- **tolerance**: Determines the amount of observations with Mahalanobis distance larger than the drawn ellipse, scaled to one.
- **locscatt**: Method for estimating the mean and covariance.
- **...**: Additional arguments passed trough.

**Value**

_no values are returned._

**Author(s)**

Peter Filzmoser, Matthias Templ

**See Also**

- ternaryDiag

**Examples**

data(coffee)
x <- coffee[,2:4]
ternaryDiag(x, grid=FALSE)
ternaryDiagEllipse(x)
## or directly:
ternaryDiag(x, grid=FALSE, line="ellipse")
ternaryDiagPoints

Add points or lines to a given ternary diagram.

Description

Low-level plot function to add points or lines to a ternary high-level plot.

Usage

ternaryDiagPoints(x, ...)

Arguments

x

Three-dimensional composition given as an object of class “matrix” or “data.frame”.  

...  

Additional graphical parameters passed through.

Value

no values are returned.

Author(s)

Matthias Templ

References


See Also

ternaryDiag

Examples

data(coffee)
x <- coffee[,2:4]
ternaryDiag(x, grid=FALSE)
ternaryDiagPoints(x+1, col="red", pch=2)
**Description**

Numerical integration via trapezoidal formula.

**Usage**

```r
trapzc(step, f)
```

**Arguments**

- `step`  
  step of the grid
- `f`  
  grid evaluation of density

**Value**

- `int`  
  The value of integral computed numerically by trapezoidal formula.

**Author(s)**

R. Talska\(<talskarenata@seznam.cz>\), K. Hron\(<karel.hron@upol.cz>\)

**Examples**

```r
# Example (zero-integral of fcenLR density)
t = seq(-4.7, 4.7, length = 1000)
t_step = diff(t[1:2])
mean = 0; sd = 1.5
f = dnorm(t, mean, sd)
f.fcenLR = fcenLR(t, t_step, f)
trapzc(t_step, f.fcenLR)
```

---

**trondelagC**

*regional geochemical survey of soil C in Norway*

**Description**

A regional-scale geochemical survey of C horizon samples in Nord-Trondelag, Central Norway

**Usage**

```r
data(trondelagC)
```
Format

A data frame with 754 observations and 70 variables

Details

- X.S_ID ID
- X.Loc_ID ID
- longitude longitude in WGS84
- latitude latitude in WGS84
- E32wgs UTM zone east
- N32wgs UTM zone north
- X.Medium
- Ag Concentration of silver (in mg/kg)
- Al Concentration of aluminum (in mg/kg)
- As Concentration of arsenic (in mg/kg)
- Au Concentration of gold (in mg/kg)
- B Concentration of boron (in mg/kg)
- Ba Concentration of barium (in mg/kg)
- Be Concentration of beryllium (in mg/kg)
- Bi Concentration of bismuth (in mg/kg)
- Ca Concentration of calcium (in mg/kg)
- Cd Concentration of cadmium (in mg/kg)
- Ce Concentration of cerium (in mg/kg)
- Co Concentration of cobalt (in mg/kg)
- Cr Concentration of chromium (in mg/kg)
- Cs Concentration of cesium (in mg/kg)
- Cu Concentration of copper (in mg/kg)
- Fe Concentration of iron (in mg/kg)
- Ga Concentration of gallium (in mg/kg)
- Ge Concentration of germanium (in mg/kg)
- Hf Concentration of hafnium (in mg/kg)
- Hg Concentration of mercury (in mg/kg)
- In Concentration of indium (in mg/kg)
- K Concentration of potassium (in mg/kg)
- La Concentration of lanthanum (in mg/kg)
- Li Concentration of lithium (in mg/kg)
- Mg Concentration of magnesium (in mg/kg)
- Mn Concentration of manganese (in mg/kg)
- Mo Concentration of molybdenum (in mg/kg)
- Na Concentration of sodium (in mg/kg)
- Nb Concentration of niobium (in mg/kg)
- Ni Concentration of nickel (in mg/kg)
- P Concentration of phosphorus (in mg/kg)
- Pb Concentration of lead (in mg/kg)
- Pb204 Concentration of lead, 204 neutrons (in mg/kg)
- Pb206 Concentration of lead, 206 neutrons (in mg/kg)
- Pb207 Concentration of lead, 207 neutrons (in mg/kg)
- Pb208 Concentration of lead, 208 neutrons (in mg/kg)
- X6_7Pb Concentration of lead (in mg/kg)
- X7_8Pb Concentration of lead (in mg/kg)
- X6_4Pb Concentration of lead (in mg/kg)
- X7_4Pb Concentration of lead (in mg/kg)
- X8_4Pb Concentration of lead (in mg/kg)
- Pd Concentration of palladium (in mg/kg)
- Pt Concentration of platinum (in mg/kg)
- Rb Concentration of rubidium (in mg/kg)
- Re Concentration of rhenium (in mg/kg)
- S Concentration of sulfur (in mg/kg)
- Sb Concentration of antimony (in mg/kg)
- Sc Concentration of scandium (in mg/kg)
- Se Concentration of selenium (in mg/kg)
- Sn Concentration of tin (in mg/kg)
- Sr Concentration of strontium (in mg/kg)
- Ta Concentration of tantalum (in mg/kg)
- Te Concentration of tellurium (in mg/kg)
- Th Concentration of thorium (in mg/kg)
- Tl Concentration of titanium (in mg/kg)
- Tl Concentration of thalium (in mg/kg)
- U Concentration of uranium (in mg/kg)
- V Concentration of vanadium (in mg/kg)
- W Concentration of tungsten (in mg/kg)
- Y Concentration of yttrium (in mg/kg)
- Zn Concentration of zinc (in mg/kg)
- Zr Concentration of zirconium (in mg/kg)

The samples were analysed using aqua regia extraction. Sampling was based on a 6.6km grid, i.e. 1 sample site/36 km².
trondelagO

Author(s)
NGU, https://www.ngu.no, transferred to R by Matthias Templ <matthiastempl@tuwien.ac.at>

References

Examples
data(trondelagC)
str(trondelagC)

trondelagO  regional geochemical survey of soil O in Norway

Description
A regional-scale geochemical survey of O horizon samples in Nord-Trondelag, Central Norway

Usage
data(trondelagO)

Format
A data frame with 754 observations and 70 variables

Details
• X.Loc_ID ID
• LITHO Rock type
• longitude longitude in WGS84
• latitude latitude in WGS84
• E32wgs UTM zone east
• N32wgs UTM zone north
• X.Medium a numeric vector
• Alt_masl a numeric vector
• LOI_480 Loss on ignition
• pH Numeric scale used to specify the acidity or alkalinity of an aqueous solution
• Ag Concentration of silver (in mg/kg)
• Al Concentration of aluminum (in mg/kg)
• As Concentration of arsenic (in mg/kg)
• Au Concentration of gold (in mg/kg)
• B Concentration of boron (in mg/kg)
• Ba Concentration of barium (in mg/kg)
• Be Concentration of beryllium (in mg/kg)
• Bi Concentration of bismuth (in mg/kg)
• Ca Concentration of calcium (in mg/kg)
• Cd Concentration of cadmium (in mg/kg)
• Ce Concentration of cerium (in mg/kg)
• Co Concentration of cobalt (in mg/kg)
• Cr Concentration of chromium (in mg/kg)
• Cs Concentration of cesium (in mg/kg)
• Cu Concentration of copper (in mg/kg)
• Fe Concentration of iron (in mg/kg)
• Ga Concentration of gallium (in mg/kg)
• Ge Concentration of germanium (in mg/kg)
• Hf Concentration of hafnium (in mg/kg)
• Hg Concentration of mercury (in mg/kg)
• In Concentration of indium (in mg/kg)
• K Concentration of potassium (in mg/kg)
• La Concentration of lanthanum (in mg/kg)
• Li Concentration of lithium (in mg/kg)
• Mg Concentration of magnesium (in mg/kg)
• Mn Concentration of manganese (in mg/kg)
• Mo Concentration of molybdenum (in mg/kg)
• Na Concentration of sodium (in mg/kg)
• Nb Concentration of niobium (in mg/kg)
• Ni Concentration of nickel (in mg/kg)
• P Concentration of phosphorus (in mg/kg)
• Pb Concentration of lead (in mg/kg)
• Pb204 Concentration of lead, 204 neutrons (in mg/kg)
• Pb206 Concentration of lead, 206 neutrons (in mg/kg)
• Pb207 Concentration of lead, 207 neutrons (in mg/kg)
• Pb208 Concentration of lead, 208 neutrons (in mg/kg)
• X6_7Pb Concentration of lead (in mg/kg)
• X7_8Pb Concentration of lead (in mg/kg)
• X6_4Pb Concentration of lead (in mg/kg)
• X7_4Pb Concentration of lead (in mg/kg)
• X8_4Pb Concentration of lead (in mg/kg)
• Pd Concentration of palladium (in mg/kg)
• Pt Concentration of platium (in mg/kg)
• Rb Concentration of rubidium (in mg/kg)
• Re Concentration of rhenium (in mg/kg)
• S Concentration of sulfur (in mg/kg)
• Sb Concentration of antimony (in mg/kg)
• Sc Concentration of scandium (in mg/kg)
• Se Concentration of selenium (in mg/kg)
• Sn Concentration of tin (in mg/kg)
• Sr Concentration of strontium (in mg/kg)
• Ta Concentration of tantalum (in mg/kg)
• Te Concentration of tellurium (in mg/kg)
• Th Concentration of thorium (in mg/kg)
• Ti Concentration of titanium (in mg/kg)
• Tl Concentration of thalium (in mg/kg)
• U Concentration of uranium (in mg/kg)
• V Concentration of vanadium (in mg/kg)
• W Concentration of tungsten (in mg/kg)
• Y Concentration of yttrium (in mg/kg)
• Zn Concentration of zinc (in mg/kg)
• Zr Concentration of zirconium (in mg/kg)

The samples were analysed using aqua regia extraction. Sampling was based on a 6.6km grid, i.e. 1 sample site/36 km².

Author(s)
NGU, https://www.ngu.no, transferred to R by Matthias Templ <matthias.templ@tuwien.ac.at>

References

Examples
data(trondelagO)
str(trondelagO)
**Description**

Youth not in employment, education or training (NEET) in 43 countries from 1997 till 2015

**Format**

A (tidy) data frame with 1216 observations on the following 4 variables.

- **country**: Country of origin
- **age**: age group
- **year**: Year
- **value**: percentage of unemployed

**Details**

This indicator presents the share of young people who are not in employment, education or training (NEET), as a percentage of the total number of young people in the corresponding age group, by gender. Young people in education include those attending part-time or full-time education, but exclude those in non-formal education and in educational activities of very short duration. Employment is defined according to the OECD/ILO Guidelines and covers all those who have been in paid work for at least one hour in the reference week of the survey or were temporarily absent from such work. Therefore NEET youth can be either unemployed or inactive and not involved in education or training. Young people who are neither in employment nor in education or training are at risk of becoming socially excluded - individuals with income below the poverty-line and lacking the skills to improve their economic situation.

**Author(s)**

translated from [https://data.oecd.org/](https://data.oecd.org/) and restructured by Matthias Templ

**Source**

OECD: [https://data.oecd.org/](https://data.oecd.org/)

**References**


**Examples**

data(unemployed)
str(unemployed)
variation

Robust and classical variation matrix

Description

Estimates the variation matrix with robust methods.

Usage

variation(x, method = "robustPivot")

Arguments

x          data frame or matrix with positive entries
method     method used for estimating covariances. See details.

Details

The variation matrix is estimated for a given compositional data set. Instead of using the classical standard deviations the miniminm covariance estimator is used (covMcd) is used when parameter robust is set to TRUE.

For method robustPivot forumala 5.8. of the book (see second reference) is used. Here robust (mcd-based) covariance estimation is done on pivot coordinates. Method robustPairwise uses a mcd covariance estimation on pairwise log-ratios. Methods Pivot (see second reference) and Pairwise (see first reference) are the non-robust counterparts. Naturally, Pivot and Pairwise gives the same results, but the computational time is much less for method Pairwise.

Value

The (robust) variation matrix.

Author(s)

Karel Hron, Matthias Templ

References


Examples

```r
data(expenditures)
variation(expenditures)  # default is method "robustPivot"
variation(expenditures, method = "Pivot")
variation(expenditures, method = "robustPairwise")
variation(expenditures, method = "Pairwise")  # same results as Pivot
```

---

**weightedPivotCoord**  
*Weighted pivot coordinates*

### Description

Weighted pivot coordinates as a special case of isometric logratio coordinates.

### Usage

```r
weightedPivotCoord(
  x,  
  pivotvar = 1,  
  option = "var",  
  method = "classical",  
  pow = 1,  
  yvar = NULL
)
```

### Arguments

- **x**: object of class ‘data.frame’ or ‘matrix’; positive values only
- **pivotvar**: pivotal variable; if any other number than 1, the data are resorted in that sense that pivotvar is shifted to the first part
- **option**: option for the choice of weights. If `option = "var"` (default), weights are based on variation matrix elements: `(1/t_1j)^pow`, if `option = "cor"`, weights are based on correlations between variable specified in `yvar` and logratios and its distribution: `|integral_0^r_j f(x) dx|`, `f(x)...` Kernel density estimator for `s_j; s_j=0` if `|r_j|<cut` otherwise `s_j=r_j`, `cut = min(#r_j>=0/#r_j, #r_j<0/#r_j`, with Gaussian Kernel function and bandwidth `h=0.05`.
- **method**: method for estimation of variation/correlation, if `option = "classical"` (default), classical estimation is applied, if `option = "robust"`, robust estimation is applied;
- **pow**: if `option = "var"`, power `pow` is applied on unnormalized weights; default is 1;
- **yvar**: if `option = "cor"`, weights are based on correlation between logratios and variable specified in `yvar`;
**Details**

Weighted pivot coordinates map D-part compositional data from the simplex into a (D-1)-dimensional real space isometrically. The relevant relative information about one of parts is contained in the first coordinate. Unlike in the (ordinary) pivot coordinates, the pairwise logratios aggregated into the first coordinate are weighted according to their relevance for the purpose of the analysis.

**Value**

<table>
<thead>
<tr>
<th>wPC</th>
<th>weighted pivot coordinates (matrix with n rows and (D-1) columns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>w</td>
<td>logcontrasts (matrix with D rows and (D-1) columns)</td>
</tr>
</tbody>
</table>

**Author(s)**

Nikola Stefelova

**References**


**See Also**

pivotCoord

**Examples**

```r
# first variable as pivotal, weights based on variation matrix
wpc_var <- weightedPivotCoord(x)
coordinates <- wpc_var$WPC
logcontrasts <- wpc_var$w

# third variable as pivotal, weights based on variation matrix, # robust estimation of variance, effect of weighting enhanced
wpc_var <- weightedPivotCoord(x, pivotvar = 3, method = "robust", pow = 2)
coordinates = wpc_var$WPC
logcontrasts = wpc_var$w

# first variable as pivotal, weights based on correlation between pairwise logratios and y
wpc_cor <- weightedPivotCoord(x, option = "cor", yvar = phd$female)
```
coordinates <- wpc_cor$WPC
logcontrasts <- wpc_cor$w

# fifth variable as pivotal, weights based on correlation between pairwise logratios
# and y, robust estimation of correlation
wpc_cor <- weightedPivotCoord(x, pivotvar = 5, option = "cor", method = "robust", yvar = phd$female)
coordinates <- wpc_cor$WPC
logcontrasts <- wpc_cor$w

---

**ZBsplineBasis**

**ZB-spline basis**

**Description**

Spline basis system having zero-integral on I=[a,b] of the L^2_0 space (called ZB-splines) has been proposed for an basis representation of fcenLR transformed probability density functions. The ZB-spline basis functions can be back transformed to Bayes spaces using inverse of fcenLR transformation, resulting in compositional B-splines (CB-splines), and forming a basis system of the Bayes spaces.

**Usage**

`ZBsplineBasis(t, knots, order, basis.plot = FALSE)`

**Arguments**

- **t**
  - a vector of argument values at which the ZB-spline basis functions are to be evaluated
- **knots**
  - sequence of knots
- **order**
  - order of the ZB-splines (i.e., degree + 1)
- **basis.plot**
  - if TRUE, the ZB-spline basis system is plotted

**Value**

- **ZBsplineBasis**
  - matrix of ZB-spline basis functions evaluated at a vector of argument values `t`
- **nbasis**
  - number of ZB-spline basis functions

**Author(s)**

J. Machalova <jitka.machalova@upol.cz>, R. Talska <talskarenata@seznam.cz>

**References**

Examples

```r
# Example: ZB-spline basis functions evaluated at a vector of argument values t
t = seq(0,20,l=500)
knots = c(0,2,5,9,14,20)
order = 4

ZBsplineBasis.out = ZBsplineBasis(t,knots,order, basis.plot=TRUE)

# Back-transformation of ZB-spline basis functions from L^2_0 to Bayes space ->
# CB-spline basis functions
CBsplineBasis=NULL
for (i in 1:ZBsplineBasis.out$nbasis)
{
  CB_spline = fcenLRinv(t,diff(t)[1:2],ZBsplineBasis.out$ZBsplineBasis[,i])
  CBsplineBasis = cbind(CBsplineBasis,CB_spline)
}

matplot(t,CBsplineBasis, type="l",lty=1, las=1,
col=rainbow(ZBsplineBasis.out$nbasis), xlab="t",
ylab="CB-spline basis",
cex.lab=1.2,cex.axis=1.2)
abline(v=knots, col="gray", lty=2)
```

zeroOut  

**Detection of outliers of zero-inflated data**

**Description**

detects outliers in compositional zero-inflated data

**Usage**

```r
zeroOut(x, impute = "knn")
```

**Arguments**

- `x`: a data frame
- `impute`: imputation method internally used

**Details**

XXX

**Value**

XXX
Author(s)

Matthias Templ

Examples

```r
### Installing and loading required packages
data(expenditures)
```
Index

* aplot
  biplot.factanal, 23
  biplot.pcaCoDa, 24
  plot.imp, 123
  plot.pcaCoDa, 125
  ternaryDiag, 149
  ternaryDiagAbline, 151
  ternaryDiagEllipse, 152
  ternaryDiagPoints, 153

* arith
  aDist, 10
  perturbation, 116

* datasets
  cancer, 26
  cancerMN, 27
  chorizonDL, 31
  efsa, 53
  expendituresEU, 60
  foodbalance, 66
  gjovik, 69
  goveexp, 73
  haplogroups, 74
  honey, 75
  laborForce, 98
  lifeExpGdp, 99
  phd, 119
  phd_totals, 120
  saffron, 134
  teachingStuff, 148
  unemployed, 160

* data
  ageCatWorld, 16
  alcohol, 17
  alcoholreg, 18
  arcticLake, 18
  coffee, 37
  economy, 51
  educFM, 52
  election, 54
  electionATbp, 56
  employment, 57
  employment2, 57
  employment_df, 58
  expenditures, 59
  GDPsatis, 66
  gemas, 67
  instw, 93
  isic32, 98
  machineOperators, 101
  manu_abs, 102
  mcad, 103
  mortality, 105
  mortality_tab, 106
  nutrients, 107
  nutrients_branded, 109
  payments, 112
  precipitation, 127
  production, 129
  rcodes, 131
  skyeLavas, 137
  socExp, 141
  trondelagC, 154
  trondelagO, 157

* hplot
  plot.imp, 123

* htest
  adtest, 13
  adtestWrapper, 14
  rSDev.test, 133

* iteration
  impCoda, 78

* manip
  addLR, 7
  addLRinv, 9
  adjust, 12
  bootnComp, 25
  ced, 28
  cenLR, 29
cenLRinv, 30
costSum, 40
impAll, 77
impKNNa, 80
impRZalr, 82
impRZilr, 84
imputeBDLs, 86
imputeUDLs, 89
is.equivalent, 97
orthbasis, 110
rdcm, 131
* math
  aDist, 10
  adjustment, 16
  pivotCoord, 121
* models
  lmCoDaX, 100
* multivariate
  clustCoDa, 33
cubeCoord, 43
daCoDa, 47
daFisher, 49
impCoda, 78
impKNNa, 80
impRZalr, 82
impRZilr, 84
imputeBDLs, 86
imputeUDLs, 89
missPatterns, 104
outCoDa, 111
pcaCoDa, 114
pfa, 117
tabCoord, 145
ternaryDiag, 149
variation, 161
weightedPivotCoord, 162
* package
  robCompositions-package, 5
* print
  print.imp, 128
  summary.imp, 144
* robust
  impCoda, 78
  variation, 161
addLR, 7, 9, 29, 30
addLRinv, 8, 9, 29, 30
aDist, 10
adjust, 12
adjustImputed (imputeBDLs), 86
adtest, 13, 15
adtestWrapper, 13, 14
ageCatWorld, 16
aggr, 105
alcohol, 17
alcoholreg, 18
all.equal, 97
arcticLake, 18
balances, 19
biomarker, 21
biplot.factanal, 23
biplot.pcaCoDa, 24, 115, 126
bootnComp, 25
cancer, 26
cancerMN, 27
ced, 28
cenLR, 9, 29, 30, 110
cenLRinv, 9, 29, 30
checkData (imputeBDLs), 86
chorizonDL, 31, 31, 33
clustCoDa, 33
clustCoDa_qmode, 36
coffee, 37
compareMahal, 38
compositionalMahal, 38
covMcd, 111, 161
cubeCoord, 43, 147
cubeCoordWrapper, 147
cubeCoordWrapper (cubeCoord), 43
daCoDa, 47
daFisher, 49
economy, 51
educFM, 52
efsa, 53
election, 54
electionATbp, 56
employment, 57
employment2, 57
employment_df, 58
INDEX

expenditures, 59
expendituresEU, 60
fcenLR, 61
fcenLRinv, 62
fcenLRp, 63
fcenLRu, 64
foodbalance, 66
GDPsatis, 66
gemas, 67
getEstimates(compareMahal), 38
gjovik, 69
gm, 72
gmean (gmean_sum), 72
gmean_sum, 72
govexp, 73
haplogroups, 74
honey, 75
ilr.2x2, 76
ilrrregression (lmCoDaX), 100
impAll, 77
impCoda, 12, 78, 79, 81, 125, 128, 144
impKNNa, 38, 80, 80, 125, 128, 144
impRZalr, 82, 85
impRZilr, 26, 78, 83, 84
imputeBDLs, 86, 88, 91
imputeUDLs, 89
ind2x2, 91
indTab, 92
instw, 93
int2x2, 94
intArray, 95
intTab, 95, 96
iprod (aDist), 10
is.equivalent, 97
isic32, 98
isomLR (pivotCoord), 121
isomLRinv (pivotCoord), 121
isomLRinvp (pivotCoord), 121
isomLRp (pivotCoord), 121
laborForce, 98
LdaClassic, 48
lifeExpGdp, 99
Linda, 48, 50
lm, 101
lmCoDaX, 100
machineOperators, 101
manu_abs, 102
mcad, 103
missPatterns, 104
mortality, 105
mortality_tab, 106
nutrients, 107
nutrients_branded, 109
orthbasis, 110
outCoDa, 111
par, 124
payments, 112
pcaCoDa, 24, 114, 126
perturbation, 116
pfa, 23, 117
phd, 119
phd_totals, 120
pivotCoord, 8, 11, 15, 29, 30, 38, 80, 110, 112, 121, 163
pivotCoordInv, 9, 29, 30
pivotCoordInv (pivotCoord), 121
plot.biomarker, 22
plot.biomarker (biomarker), 21
plot.clustCoDa (clustCoDa), 33
plot.clustCoDa_qmode (clustCoDa_qmode), 36
plot.estimates (compareMahal), 38
plot.imp, 123
plot.mahal (compareMahal), 38
plot.outCoDa (outCoDa), 111
plot.pcaCoDa, 24, 115, 125
plot.smoothSpl, 126
powering (perturbation), 116
precipitation, 127
predict.daFisher (daFisher), 49
print.adtestWrapper (adtestWrapper), 14
print.biomarker (biomarker), 21
print.coord (coord), 41
print.daFisher (daFisher), 49
print.estimates (compareMahal), 38
print.imp, 128
print.outCoDa (outCoDa), 111
print.pcaCoDa, 115
print.pcaCoDa (pcaCoDa), 114
print.replaced (imputeBDLs), 86
production, 129
pTab, 130
QdaClassic, 48
QdaCov, 48
rcodes, 131
rdcm, 28, 131, 132
robCompositions
  (robCompositions-package), 5
robCompositions-package, 5
robilrregression (lmCoDaX), 100
rSDev, 132, 134
rSDev.test, 133
saffron, 134
SDev, 136
skyeLavas, 137
smoothSplines, 138, 140
smoothSplinesVal, 140
socExp, 141
stats, 142
summary.adtestWrapper (adtestWrapper), 14
summary.biomarker (biomarker), 21
summary.daFisher (daFisher), 49
summary.imp, 144
summary.pcaCoDa, 115
summary.pcaCoDa (pcaCoDa), 114

tabCoord, 46, 145
tabCoordWrapper, 46
tabCoordWrapper (tabCoord), 145
teachingStuff, 148
ternary, 150
ternaryDiag, 149, 151–153
ternaryDiagAbline, 151
ternaryDiagEllipse, 152
ternaryDiagLines (ternaryDiagPoints), 153
ternaryDiagPoints, 153
trapez, 154
trondelagC, 154
trondelagO, 157
unemployed, 160
variation, 161
weightedPivotCoord, 162
ZBsplineBasis, 164
zeroOut, 165
zeroPatterns (missPatterns), 104